3

Z MIKROKOMPUTEREM NA TY

Bajtek

NR INDEKSU 353965
PL ISSN 0860 – 1674

MIESIĘCZNY DODATEK DO "SZTANDARU MŁODYCH" NR 3(15) MARZEC 1987 CENA 100ZŁ



CO PISZCZY POD KLAWIATURĄ?

GEOS GONI MACINTOSHA

„DUSZEK" NA ATARI

HAZARDZIŚCI

FANTASTIC VOYAGE

POZA PROGRAMEM

Fizyka jest szczególną z nauk, gdyż jako jedna z nielicznych posiada zdolność całościowego objaśniania świata. Jest źródłem wielu pojęć mających decydujące znaczenie we wszystkich badaniach przyrody, w tworzeniu jednolitego, obowiązującego nas wszystkich obrazu rzeczywistości. Przykładami takich pojęć są: energia, entropia, czasoprzestrzeń, komplementarność, symetria, pole, chaos... Dlatego zachęcamy wszystkich ambitnych Czytelników „Bajtka" do studiowania fizyki. Publikowana obok rozmowa z prof. Andrzejem Kajetanem Wróblewskim – członkiem Polskiej Akademii Nauk i jego kolegami z Wydziału Fizyki Uniwersytetu Warszawskiego pokazuje, że istnieje możliwość zgłębiania praw fizyki i jednocześnie rozwijania swoich zainteresowań komputerowych. Warto z tej możliwości skorzystać!

– Co z okresu edukacji w szkole jest najważniejsze dla przyszłego uczonego? – takie pytanie zadałem niedawno w Moskwie wybitnemu fizykowi radzieckiemu, członkowi Akademii Nauk ZSRR, prof. Witalijowi Ginzburgowi. Kandydatów do studiowania fizyki (i nie tylko) zachęcam do przeczytania kilku rad uczonego, który współtworzył spory fragment współczesnej nauki.

– Po pierwsze – odpowiedział prof. Ginzburg – szkoła powinna dawać umiejętność pisania bez błędów językiem literackim, jasnego przekazywania swoich myśli. Pewne umiejętności daje w tej mierze i uczelnia, na przykład w trakcie przygotowywania prac przejściowych i dyplomowych. Ale podstawy powinna dawać szkoła. Mnie nie dała. W 1934 roku, na drugim semestrze studiów-, przeprowadzono u nas na Uniwersytecie Moskiewskim dyktando, i połowa studentów, łącznie ze mną, otrzymała ocenę niedostateczną. Wprowadzono potem zajęcia z języka rosyjskiego, ale niewiele one mi dały. Muszę do dziś często zaglądać do słownika, myśleć nad budową nawet prostych zdań, sprawdzać to co napisałem. Staram się więcej dawać do przepisania na maszynie. Ale czyż maszynistka ma obowiązek poprawiania błędów i czyż można zdawać się na to?

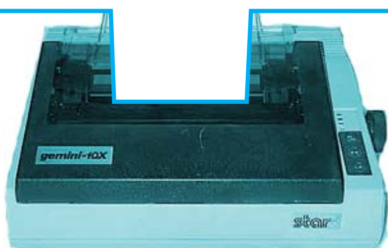
Po drugie – szkoła powinna zapewnić automatyzm w zakresie elementarnej matematyki. Mam na myśli umiejętność szybkiego liczenia, nawyki w zakresie arytmetyki, algebry, trygonometrii, korzystania z komputerów osobistych. To też osiąga się treningiem. Uczniów nudzi odmienianie i koniu-gowanie, nauka zasad gramatyki, rozwiązywanie wiele razy prawie jednakowych zadań i dokonywanie przekształceń, które już ze swej istoty są oczywiste. Tak się złożyło, że do szkoły podstawowej chodziłem tylko 4 lata i zdając na studia musiałem w ciągu 3 miesięcy opanować materiał z wielu lat. Dlatego, powiedzmy, rozwiązałem 100 zadań zamiast 1000, które rozwiązałbym w szkole. Rezultat tego – brak nawyków automatyzmu – odczuwam przez całe życie. Dlatego radzę nie oszczędzać czasu poprzez zmniejszanie liczby zadań, przykładów, ćwiczeń. To fałszywa oszczędność.

Trzecie – nie wymaganie a życzenie – jeszcze w szkole należy opanować język angielski. Do drugiej wojny światowej dominował w fizyce język niemiecki, teraz nawet w RFN i NRD czasopisma fizyczne zamieszczają artykuły w przeważającej części po angielsku. Właśnie język angielski stał się międzynarodowym językiem nauki i koniecznością jest znać go. Tracić wiele czasu na uczelni aby nauczyć się języka jest po prostu głupie i nieracjonalne.

Czwarta i ostatnia uwaga – należy wyjść poza program szkolny, zetknąć się chociaż trochę ze współczesnym stanem nauki. Szkoła powinna zapewnić uczniom taką możliwość. A jeśli nie zapewni? Trzeba samemu szukać w czasopismach popularno-naukowych, w trakcie publicznych wykładów itp. Usłyszę pewnie: „A skąd wziąć czas, skoro tak dużo zadają do domu?" Jest to i prawda, i nieprawda zarazem. Na to, co rzeczywiście nas interesuje czas zawsze się znajdzie. Nie można nauczyć się pływać nie wchodząc do wody!

Tyle akademik Ginzburg. Mam nadzieję, że i lektura „Bajtka" jest dla wszystkich przyszłych studentów okazją do takiego pierwszego zetknięcia się z problemami współczesnej nauki, przede wszystkim informatyki, ale nie tylko. Po to m.in. wprowadziliśmy rubrykę „Nie tylko komputery". Po-wodzenia na maturze i podczas egzaminów na studia!

Waldemar Siwiński



Szanowny Panie Redaktorze

Na wstępie pragnę wyrazić moją wdzięczność płynącą z tego, iż postanowiliście pomóc osobom niepełnosprawnym posiadającym własny komputer. Krótko o sobie.

Dnia 30.07.1982 r. po skoku do wody doznałem urazu kręgosłupa szyjnego z porażeniem czterokończynowym. W praktyce wygląda to tak, iż od tego dnia jestem wyłącznie „mózgowcem". Przedtem byłem muzykiem. Grałem czynnie na instrumentach klawiszowych, dużo komponowałem. Moją drugą pasją były szachy. Bardzo długo nie mogłem pogodzić się z sytuacją w jakiej się znalazłem. Moi przyjaciele walczyli długo o to, by wyciągnąć mnie z „dołka".

Nie mam szans na chociażby częściowe wyleczenie. Jestem przykuty do łóżka i od piersi w dół nie czuję nic. Mogę wykonywać bardzo ograniczone ruchy rękoma (bez palców). Moi rodzice razem z przyjaciółmi kupili dla mnie — mój wymarzony komputer — AMSTRAD 464 z zielonym monitorem i odtąd zaczęła się moja rozpacz. Nie mam prawie żadnego oprogramowania do mojej „Jedynej Radości". Nie mam instrukcji po polsku.

Mój serdeczny przyjaciel załatwił mi dwa programy i na tym kończą się jego możliwości. Napisałem do f-my „Amico". Przysłano mi wykaz programów jakie oni kopiują. Niestety nie jestem w stanie zapłacić takiej sumy za te programy (mam rentę w wysokości 8060 - sic!). Chciałbym teraz wyjaśnić jak obsługuję „moją perełkę". Do przedramion mam przypięte skórzanymi paskami aluminiowe rurki, wygięte na końcu i zakończone gumowymi kulkami. Cóż mogę jeszcze napisać. Komputer to dla mnie wszystko. Chciałbym uczyć się języków przy jego pomocy grać w szachy i nade wszystko wrócić do komponowania muzyki.

Cieszę się Bajtku z powodu tej iskierki nadziei, jaką nabrałem po przeczytaniu 10 numeru wspaniałego pisma.

Grzegorz S.
(Nazwisko i adres znane redakcji)

Pragnę podzielić się z Redakcją pomysłem, który — chwała niech będzie Archimedesowi — przyszedł mi do głowy w wannie.

Proponuję mianowicie, by Bajtek wystąpił do redakcji popularnych pism informatycznych z całego świata (z wyłączeniem pism typu Atari User, czy Am-strad Magazine) z inicjatywą utworzenia międzynarodowego kolegium redaktorów tych pism, przyznającego (wzorem Oscara, czy Cezara) dorocznych nagród „Złotego Joysticka", albo może „Złotej Myszy". Winny być one przyznawane w kategorii sprzętu (oddzielnie komputery domowe i osobiste), programów użytkowych (również dla HC i PC), oraz gier. W tej ostatniej grupie należałoby, moim zdaniem dokonać dodatkowego podziału: program, jako całość, grafika, oprawa muzyczna, oraz scenariusz.

Leszek Robakowski
ul. Pionierska 6 m 1
38-700 Ustrzyki D.

Uprzejmie dziękuję za opublikowanie adresu naszego Klubu, ten dzień bowiem wyznacza początek bujnej działalności. Wraz z setkami listów, które otrzymaliśmy, pytaniami i propozycjami krystalizował ostateczny profil OMIC-u. U podstaw było to luźne stowarzyszenie użytkowników AMSTRADÓW, którego funkcjonowanie oparte było na demokratycznych zasadach niezobowiązującej współpracy. Dzięki Wam, obecnie jesteśmy silną wpływową organizacją, o kilkudziesięciu członkach, statusie, kilkunastu komputerach i przede wszystkim o wielkim zapale do pracy!

Klub Komputerowy OMIC
Prezes
Aleksander Matczewski

WYBIERZ SAM

„BAJTEK" — MIESIĘCZNY DODATEK DO „SZTANDARU MŁODYCH"
ADRES: 00-687 Warszawa, ul. Wspólna 61. Telefon 21-12-05

Przewodniczący Rady Redakcyjnej: Jerzy Domański-redaktor naczelny „Sztandaru Młodych".

ZESPÓŁ REDAKCYJNY: Waldemar Siwiński (z-ca redaktora naczelnego „SM" - kierownik zespołu „Bajtka"), Roman Poznański (z-ca sekretarza redakcji „SM" - sekretarz zespołu „Bajtka"), Krzysztof Czernek, Sławomir Gajda (red. techniczny), Andrzej Gogolewski, Andrzej Kowalewski, Andrzej Podulka, Sławomir Polak, Wanda Roszkowska (opr. graficzne), Kazimierz Treger, Marcin Waligórski, Roman Wojciechowski.

Klany redagują:
Commodore - Klaudiusz Dybowski, Michał Silski; Amstrad-Schneider - Tomasz Pyć, Sergiusz Wolicki; Spectrum - Konrad Fedyna, Michał Szuniewicz; Atari — Wiesław Migut, Wojciech Zientara.

Fotoskład — Tadeusz Olczak,
montaż offsetowy — Grażyna Ostaszewska,
korekta— Maria Krajewska, Ewa Mowińska

WYDAWCA: RSW „Prasa-Książka-Ruch" Młodzieżowa Agencja Wydawnicza, al. Stanów Zjednoczonych 53, 04-028 Warszawa. Telefony: Centrala 13-20-40 do 49, Redakcja Reklamy 13-20-40 do 49 w. 403, 414. Cena 100 zł.
Skład techniką CRT-200, przygotowalnią offsetowa i druk: PRASOWE ZAKŁADY GRAFICZNE RSW PRASA-KSIĄŻKA-RUCH" w Ciechanowie, ul. Sienkiewicza 51.
Zam. nr 1432/86, nakład 250000 egz., P101



Bajtek

Rozmowa z prof. Andrzejem Kajetanem Wróblewskim — dziekanem Wydziału Fizyki Uniwersytetu Warszawskiego oraz dr. hab. Ryszardem Kutnerem i dr. Romanem Szwedem — organizatorami specjalności „fizyka komputerowa" na tym wydziale.



HAZARDZIŚCI

— Panie Profesorze, czy ma Pan w domu komputer?

Andrzej Wróblewski: — Oczywiście, że mam. Używam ich w pracy naukowej i bawię się nimi, kiedy tylko pozwala mi na to czas.

— A inni polscy fizycy?

AW: — Większość szybko zaakceptowała komputer. Np. profesor Jerzy Pniewski, senior naszego wydziału, który już od trzech lat jest na emeryturze, uwielbia po prostu pracę z komputerami i rozwiązuje na nich w domu większość problemów, którymi się zajmuje. Oczywiście młodsi koledzy częściej na ogół niż moje pokolenie interesują się programowaniem. Mają na to więcej czasu. Niektórzy tak mocno wciągają się w zabawę z komputera, że zapominają na pewien czas o fizyce... Na szczęście są to wyjątki. Znacznie większej liczbie właśnie komputer dał możliwość opracowania ciekawej pracy magisterskiej czy doktorskiej, czy nawet — jak w przypadku doktora Ryszarda Kutnera — habilitacji.

— Większość uczniów szkoły średniej nie kojarzy sobie jednak fizyki z komputerami. Jest to dla nich raczej dziedzina wiedzy opisywana przez precyzyjne wzory analityczne. Do otrzymania wyników zadania po kilku podstawieniach i przekształceniach algebraicznych potrzebny jest co najwyżej kalkulator programowalny. Czy zatem zastosowanie komputerów w fizyce nie jest odstępstwem od opisu dokonywanego poprzez „czyste wzory" i przyznaniem się do tego że za jego pomocą nie uda się opisać przyrody?

AW: — Nie można przez „czystą fizykę" rozumieć tylko tego, co da się opisać wzorami takimi, jakie znamy z mechaniki Newtona. Istnieje wiele praw fizycznych, które nie są prawami deterministycznymi lecz statystycznymi. Opisują one m.in. szereg zjawisk zachodzących w mikroświecie. Nawet tak prostych efektów jak przejście elektronu przez kryształ nie da się opisać wzorem, który ściśle przewidziałby, co się z tym elektronem stanie.

Jest jednak trochę racji w tym, co pan powiedział. Mogę podać przykład ze swojej dziedziny. Oddziaływania na poziomie najmniejszych cząste-czek-kwarków i partonów nie doczekały się w ogóle jeszcze spójnej teorii. W tej sytuacji problemów z tej dziedziny nie rozwiązuje się analitycznie lecz wyłącznie za pomocą komputerów i to tych największych. Przydałyby się zresztą jeszcze potężniejsze, nawet CRAY-a.

Całym wielkim działem fizyki, który nie może się dziś obejść bez komputerów, jest także fizyka statystyczna.

— Tam wkroczyły one w pierwszej kolejności?

AW: — Tak, w metodach statystycznych fizyki jądrowej, a dokładnie podczas prac nad projektem „Manhattan" budowy pierwszej bomby atomowej. Trzeba było wówczas odpowiedzieć na pytanie: jak zachowują się neutrony w reaktorze jądrowym? Zderzenia tych neutronów są sprężyste. Ale neutronów tych w reaktorze jest ogromna, nieznana zresztą dokładnie liczba. Dlatego właśnie postanowiono posłużyć się statystyczną metodą MonteCarlo której jednym z twórców był polski matematyk Stanisław Ułam.

— Nazwa kojarzy się z hazardem.

AW: — Bo to jest coś w rodzaju hazardu. Metoda ta jest jednak szeroko stosowana do dziś nie tylko do obliczeń lecz także przy planowaniu eksperymentów podczas tzw. akceptacji aparatury.

— Przykłady podane przez pana Profesora dotyczą mikroświata. A w skali makro?

AW: — Tu także często piękne równania opisujące zachowanie się obiektów nie dają się rozwiązać analitycznie i trzeba sięgać po komputer. W skali „super makro" komputery też są pożyteczne. Oddały duże usługi kosmologii przy modelowaniu wczesnych etapów rozwoju wszechświata.

— I wszystko to o czym mówiliśmy jest fizyką komputerową?

AW: — Oczywiście, że nie. Do niedawna całą fizykę podzielić można było na teoretyczną i doświadczalną. W pierwszej opisujemy świat za pomocą formuł, czystej matematyki, w drugiej badamy go poprzez doświadczenia. Natomiast fizyka komputerowa charakteryzuje się tym, że komputer nie pomaga w przeprowadzeniu rachunków lub eksperymentu, lecz jest jedynym narzędziem naukowca. Zgodność wyników z doświadczeniem sprawdza się korzystając z dorobku innych.

— Istnieje wszakże podział fizyki na wiele działów nie ze względu na metody, lecz przedmiot badań. Jest biofizyka, astrofizyka, fizyka ciała stałego, optyka itp. Czy fizyk komputerowy musi umieć poruszać się w każdym z tych działów?

AW: — Raczej można mówić np. o fizyce teoretycznej ciała stałego, fizyce doświadczalnej ciała stałego i fizyce komputerowej ciała stałego. Pojawiła się zatem nowa metoda badawcza, tradycyjny zaś podział na działy pozostał. Fizyka komputerowa to część czegoś, co dzisiaj nazywamy „wiedzą komputerową" — „computer science".

— Zamierzają panowie w waszym wydziale powołać pierwszą w naszym szkolnictwie wyższym specjalizację „fizyka komputerowa". W świetle tego, co dotąd powiedzieliśmy nie tylko jednak absolwenci tej specjalności posługiwać się będą komputerami — jest to narzędzie każdego fizyka.

Ryszard Kutner: Wszyscy studenci fizyki mają zajęcia z komputerami. Na II semestrze I roku uczą się posługiwania mikrokomputerem profesjonalnym IBM PC XT, a także pisania programów w Fortranie, BASIC-u, PASCAL-u itd. Przez cały III rok natomiast prowadzone są zajęcia z metod numerycznych fizyki.

Specjalność fizyka komputerowa będzie oczywiście bardziej niż inne nasycona metodami numerycznymi. Z pomocą komputera studenci rozwiązywać będą naprawdę skomplikowane problemy z różnych dziedzin fizyki. Chcemy, by nasz absolwent był ekspertem nie tyle w dziedzinie kompilatorów czy syntaktyki języków lecz praktycznego wykorzystania komputerów, by potrafił bez trudu zastosować je nawet poza fizyką np. do sterowania robotem, czy linią produkcyjną. Jesteśmy przekonani, że nasi absolwenci nie tylko będą mogli uprawiać fizykę, ale także znaleźć sprawiającą im zadowolenie pracę wszędzie tam, gdzie są w użyciu komputery. Metody komputerowe np. optymalizacyjne, symulacyjne itd. stosowane w fizyce mają charakter bardziej uniwersalny, mogą być przeniesione do wielu dziedzin medycyny, ekonomii, socjologii, technologii.

— A sprzęt? Taki program zajęć z komputera musi zakładać jego pełną dostępność dla wszystkich studentów.

Roman Szwed: Bez sprzętu w ogóle nie moglibyśmy mówić o fizyce komputerowej. Uczenie jej na „sucho" byłoby oczywistym nonsensem. Dziś mamy już ok. 50 IBM-ów PC XT i AT, 3 SM-y oraz końcówki znajdującego się w Świerku dużego komputera CYBER i BASF-ów z ośrodka informatyki UW. Te ostatnie są już maszynami o dużych możliwościach z 8 megabajtami pamięci operacyjnej. Chcielibyśmy dokupić jeszcze np. komputery typu VAX i płytki wektorowe do IBM PC AT.

RK: — Na świecie tworzy się centra przy wielkich komputerach i równolegle stawia na mikroin-formatykę. Tą ostatnią drogą idą mniejsze ośrodki, takie jak nasz. Nie są one jednak bez szans zważywszy na olbrzymi postęp w dziedzinie konstrukcji mikrokomputerów. Już dziś myśli się o wyposażeniu ich w comact dyski komputerowe, które zapewnią im pamięć rzędu gigabajtów, wprowadza tran-sputery powodujące, że IBM PC AT stają się komputerem wektorowym. Niedługo zapewne dorównają swoimi możliwościami dzisiejszym potężnym CRAY-om.

AW: — Dobrze byłoby dysponować i mikrokomputerami i potężnym centrum obliczeniowym, ale to co mamy też oddziaływuje na wyobraźnię młodych ludzi. Potrafią ją pobudzić przecież i komputery domowe.

— I tych nie brakuje chyba na wydziale fizyki. Macie państwo jak słyszałem, pracownię dydaktyki komputerów szkolnych. Czy można zrewolucjonizować przy ich pomocy nauczanie fizyki w szkołach?

AW: — Już 10-15 lat temu widziałem jak można wykładać za pomocą komputera. W jednym z niemieckich uniwersytetów komputer pokazywał rozwiązania równań Keplera przy zmianie np. stałej grawitacji. Studenci mogli obserwować jaki byłby nasz układ słoneczny, gdyby ją podwoić. Taka metoda pokazywania świata i jego praw weszła już do o wiele bardziej zaawansowanej fizyki. Właśnie tłumaczony jest na język polski podręcznik „Mechanika kwantowa w obrazkach". Są w nim wzory, ale także „filmy" komputerowe pokazujące takie zjawiska jak np. przechodzenie cząstki przez barierę potencjału, rozpraszanie itp.

Dydaktyka komputerowa jest fascynująca. Można dzięki niej np. zobaczyć, jak wyglądałby świat, gdyby inna była prędkość światła, być niemalże bogiem stworzonego przez siebie świata.

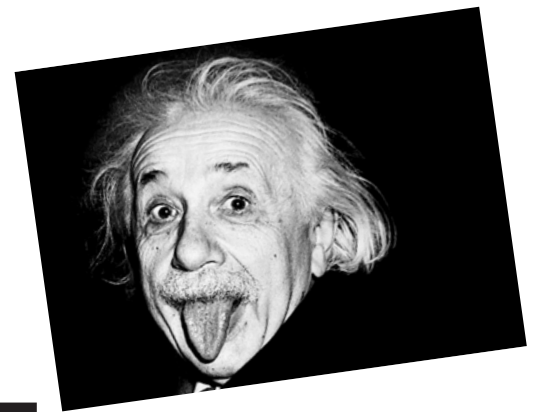
— Fizycy mogliby przybliżyć taką dydaktykę polskiej szkole np. przez opracowywanie dla niej odpowiednich programów.

RK: — Niełatwo przekonać naukowca, by zajął się dydaktyką, ale czasem udaje się to zrobić. Nie wolno nam jednak przestraszyć nauczycieli, którzy wiedzą dobrze, że komputery pojawić się muszą zamiast czegoś. Programy szkolne są już i tak przeładowane.

A.W. Prowadzimy w szkołach kółka komputerowe. Sądzę, że one właśnie mogłyby opracowywać programy dydaktyczne. To może być naprawdę bardziej pasjonujące od gier.

Sinclair zbudował też swój komputer nie z myślą o grach, lecz przeciwnie — gry na „Spectrum" służyć miały popularyzacji komputera, który wykorzystać można w różnych dziedzinach życia. Dla mnie komputer jest interesujący w połączeniu z fizyką. Tych przyszłych maturzystów, którzy mają podobne zainteresowania zachęcić mogę jedynie do podjęcia studiów na naszym wydziale.

Rozmawiał
Grzegorz Onichimowski



POLSKI DYSK

W roku przyszłym potrzeba będzie w kraju około 3.5 mln dysków elastycznych, natomiast w 1990 roku już prawie 6.5 mln sztuk. Takie sa ostrożne szacunki biorące pod uwagę przede wszystkim dyskietki 8 i 5.25 cala, pomijające nawet coraz bardziej popularne, małe — ujęte w sztywna obudowę — dyski 3.5 calowe.

Jest to także skala problemu, który wziął na swe barki warszawski „Meral" — Przedsiębiorstwo Projektowe i Modernizacji Przemysłu Automatyki i Aparatury, będący w praktyce koordynatorem całego „dyskietkowego" przedsięwzięcia. Spójrzmy jednak od końca, czyli... podwykonawcy. Taka jest bowiem na razie rola Zakładów Włókien Chemicznych „STILON", które produkować mają w przyszłości owe polskie dyski elastyczne

Gorzowski „STILON" kojarzy się z taśmami magnetofonowymi. Z rozwojem ich produkcji wiążą się także pierwsze kroki poczynione na „komputerowej drodze" przez fabrykę w początkach lat siedemdziesiątych. Kupując wówczas od „AGFY-GEVERT" licencję na nowoczesne taśmy szpulowe i kasety zakład otrzymał też technologię i urządzenia do produkcji taśmy do maszyn cyfrowych. Wytwarzano ją w ilości kilkudziesięciu tysięcy kilometrów rocznie. W końcu lat siedemdziesiątych produkcję taśmy komputerowej musiała jednak fabryka wstrzymać, zabrakło dewiz na import surowców oraz zakup nowych zachodnich urządzeń do kontroli jakości taśmy, do starych bardzo już zużytych — nie wytwarzano już po prostu za granicą części zamiennych.

W latach siedemdziesiątych zapoczątkowano w „STILONIE" prace nad nowymi lakierami magnetycznymi, które doprowadzić miały do powstania taśm wyższej jakości zarówno magnetofonowych jak i cyfrowych; Zakład raz po raz ponawiał staranie o ponowne uruchomienie produkcji, zwłaszcza że taśma cyfrowa nadal ma w naszych warunkach pewne zalety, odznaczając się dużą pojemnością magazynowania

Diametralny zwrot w sytuacji przyniosły ostatnie lata: masowa produkcja komputerów z pamięciami na dyskach twardych, elastycznych także „winchester" i pamięci kasetowe. Jasnym stało się, że nie opłaca się inwestować dewiz (których zresztą sam „STILON" pozbawiony praktycznie eksportu me ma) w odradzanie przestarzałe] produkcji taśmy cyfrowej. Pewne nadal niezbędne ilości lepiej kupić za granicą. Podobnie jak już wcześniej zrobili na przykład Bułgarzy (aczkolwiek drogą zakupu licencji) postawiliśmy na własne dyski elastyczne. Technologię ich produkcji opracować ma dla „STILONU" — zgodnie z rolą wyznaczoną przez „MERAL" — Filmowy Ośrodek Badawczo-Rozwojowy „TECHFILM" w Warszawie, który powinien zakończyć prace w połowie roku.

W tym miejscu zaczyna się rola gorzowskiego „STILONU", który korzystając z dotychczasowych doświadczeń, ma zamiar jak najszybciej uruchomić produkcję polskich dyskietek, tu też jednak zaczynają się wielkie niewiadome, o których przesądzi zapewne w najbliższych miesiącach podpisanie umowy między „MERALEM", a ministerstwem czyli Urzędem ds. Nauki i Postępu Technicznego. Do wyboru są w praktyce dwie drogi albo zdecydować się na zakup urządzeń do produkcji dysków elastycznych za granicą, albo też próbować wykonać je w kraju. Pierwsze pozwoliłoby odrobinę nadgonić stracony dystans. Kupując na początek urządzenia do obróbki importowanych „placków" czyli odpowiednio wyciętej i powleczonej warstwą magnetyczną folii, „STILON" mógłby dostarczyć pierwsze dyski już po kwartale — góra po półroczu. Opanowanie owego drugiego etapu produkcji (szlifowanie, polerowanie, przygotowanie kopert — laminowanie, zgrzewanie itp.) umożliwiłoby też sprawne i szybkie przejścia do oblewu folii we własnym zakresie (po sprowadzeniu odpowiednich maszyn). Drugie rozwiązanie, czyli zdanie się na własne siły przy konstruowaniu urządzeń do oblewu i lakierowanie, a także obróbki oznacza dalszych 3 — 4 lata pracy w nailepszym razie otrzymanie polskich dyskietek w 1990 roku.

Czy oznacza to konieczność wyłożenia kolejnego miliona dolarów? Spora cześć tej sumy zostanie zaoszczędzona bowiem niepotrzebny stanie się bardzo kosztowny import gotowych dysków elastycznych. Z drugiej strony tysiące dolarów na sprowadzenie maszyn do obróbki „placków" w kraju |uż przeznaczono, urządzenia zamówiło wrocławskie „ELWRO", które obok komputerów chce też wytwarzać do nich pamięci.

Niewątpliwie jest to godna uznania troska zakładu o odbiorcę swego sprzętu. Pozostaje |ednak pytanie czy stać nas właśnie dziś na rozpraszanie środków dewizowych? Czy instalować maszyny obróbcze w „ELWRO", by stale (albo bardzo długo) importować do nich folię magnetyczną, czy też może lepiej) postawić je od razu u docelowego producenta — w „STILONIE"? Uniknięto by ponownego zakupu tych samych urządzeń dla gorzowskiej fabryki, zaoszczędzono na szybkie sprowadzenie maszyn do oblewu folii, a jednocześnie umożliwiono załodze „STILONU" opanowanie połowy operacji produkcji dyskietki. Załodze — podkreślam — mającej już spore doświadczenie w tej dziedzinie

Jeszcze czas podjąć |est w tej sprawie decyzje. Dotychczasowe przy tym doświadczenia uczą, że wszelkie prowizorki i połowiczne rozwiązania wyłącznie oddalają nas nie tylko od czołówki ale i od zwykłych przeciętniaków na komputerowym rynku.

Andrzej Cudak

TRANSFORMACJE

PRZETWARZANIE FIGUR GEOMETRYCZNYCH

Poniższy program zrodził sie z potrzeby chwili. Przy okazji pisania artykułu „ZOSTAŃ FILMOWCEM” (zob. bieżący numer Bajtka) zainteresował mnie problem automatycznej generacji obrazów animowanych. Chodziło oczywiście o zaoszczędzenie czasu przy tworzeniu filmów.

Schemat rozwiązania problemu znalazłem m.in. w książce I.O. Angella „Wprowadzenie do grafiki komputerowej”. Z kolei publikacja J.R. Truckenbroda „Computer-Assisted Education In The Visual Arts” naprowadza na szereg innych ciekawych zastosowań tej techniki programistycznej.

Pomysł polega na przedstawieniu pewnych wybranych (najczęściej skrajnych) stadiów ruchu obiektu w postaci łamanych, a następnie na ich podstawie wyliczenia i wykreślenia żądanej ilości „stadiów pośrednich”.

Do zapisania programu wybrałem Logo, ze względu na kilka jego ważnych zalet. Po pierwsze grafika żółwia umożliwia posługiwanie się układem współrzędnych prostokątnych. Po drugie, program w Logo zawsze stanowi strukturę otwartą; użytkownik ma swobodę manipulowania procedurami programu oraz jego rozszerzenia, co jest szczególnie ważne przy tworzeniu grafiki.

Program stanowi zbiór procedur narzędziowych, pozwalających na tworzenie i transformację wyżej opisanych łamanych. Łamaną opisujemy przy pomocy listy, stanowiącej ciąg punktów. Punkt z kolei, zgodnie z konwencją Logo, zapisany jest w formie listy dwuelementowej, zawierającej jego współrzędne.

Opiszmy działanie procedur programu.

PRZEOBRAZ :L1 :L2 :Usuwanie :IleKroków

Utworzenie i wykreślenie na ekranie łamanych :L1 i :L2 oraz wszystkich faz pośrednich, których ilość wyznacza parametr :IleKroków. Parametr :Usuwanie może przyjąć wartość „T lub „N i określa czy fazy pośrednie rysunku mają być ukazywane kolejno czy wszystkie naraz (wszystkie zamieszczone ilustracje powstały przy osuwanie = „N). Przykładem wywołania może być:

PRZEOBRAZ :Kwadrat1 :Kwadrat2 „N 10

RYSUJ :StanPióra :L

Narysowanie łamanej :L. Stan pisaka żółwia określa parametr :StanPióra. Procedurę wywołujemy np.

RYSUJ [PENDOWN] [[40 40] [40 40]]

PRZESUŃ :L :Wektor

Przesunięcie wszystkich punktów łamanej o nazwie :L o wektor określony przez drugi parametr. Należy zwrócić uwagę, że :L oznacza nazwę zmiennej a nie jej wartość, np.

PRZESUN „Kwadrat1 [60 0]

KASUJ :L

Nadaje zmiennej o nazwie L wartość listy pustej (pustej łamanej).

NOTUJ :L

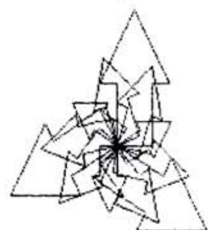
Dołączenie do łamanej o nazwie :L punktu wskazanego przez aktualną pozycję żółwia. Oto przykład użycia w powiązaniu z procedurą KASUJ:

KASUJ „Kwadrat1 HOME
NOTUJ „Kwadrat1
REPEAT 4 [FD 40 RT 90 NOTUJ „Kwadrat1]
RYSUJ [PD] :Kwadrat1

Zamieszczone ilustracje (czytelnik na pewno domyśli się, jak je we własnym zakresie odtworzyć) ukazują zastosowanie programu nie tylko w celu uzyskania efektów animacji, lecz także — zgodnie z propozycją J.R. Truckenbroda — do produkcji ciekawych grafik komputerowych, mogących służyć jako studium rytmu i proporcji w rysunku.

Marcin Waligórski

```
MAKE "KW2 [[60 -68] [100 -68] [1
00 -88] [80 -88] [80 -68]]
MAKE "KW1 [[-80 -88] [-80 -68] [
-60 -68] [-60 -88] [-80 -88]]
```



```
TO PRZEJDZ :Lista
IF EMPTYP :Lista [STOP]
SETPOS FIRST :Lista
PRZEJDZ BF :Lista
END

TO KASUJ :Zmienna
MAKE Zmienna []
END

TO NOTUJ :Zmienna
MAKE :Zmienna LPUT LIST XCOR
YCOR THING :Zmienna
END

TO PRZESUN :Zmienna :Wektor
MAKE :Zmienna PRZES THING
:Zmien na FIRST :wektor LAST
:Wektor
END

TO PRZES :Lista :DX :DY
IF EMPTYP :Lista [OP []]
MAKE "Punkt1 FIRST :Lista
MAKE "NPunkt LIST :DX + FIRST
Punktl :DY + LAST :Punkt1
OP SENTENCE BL LIST :NPunkt []
PRZES BF :Lista :DX :DY
END

TO TRANSF :Lista1 :Lista2
:IleKrokow
IF OR EMPTYP :Lista1
EMPTYP :Lista2 [OP []]
MAKE "Punkt1 FIRST :Lista1
MAKE "Punkt2 FIRST :Lista2
MAKE "Xl FIRST :Punkt1
MAKE "Y1 LAST :Punkt1
MAKE "X2 FIRST :Punkt2
MAKE "Y2 LAST :Punkt2
MAKE "NPunkt LIST
SUM :X1 (:X2 - :X1) /
:IleKrokow SUM :Y1 (:Y2 - :Y1)
/ :IleKrokow
OP SENTENCE BL LIST :NPunkt []
TRANSF BF :Lista1
BF :Lista2 :IleKrokow
END

TO RYSUJ :StanPiora :Lista
IF EMPTYP :Lista [STOP]
PENUP
SETPOS FIRST :Lista
RUN :StanPiora PRZEJDZ :Lista
END

TO PRZEOBRAZ :Lista1 :Lista2
:Usuwanie :IleKrokow
MAKE "Opcja MEMBERP :Usuwanie
[Tt]
RYSUJ [PENDOWN] :Lista1
MAKE "FigPoprz :Lista1
MAKE "Krok 0
REPEAT :IleKrokow
[MAKE "Krok :Krok + 1
MAKE "FigNast TRANSF :Lista1
:Lista2 :IleKrokow / :Krok
IF :0pcja [RYSUJ [PENERASE]
:FigPoprz] [] RYSUJ [PENDOWN]
:FigNast
MAKE "FigPoprz :FigNast]
END
```

polski LOGO NA ATARI

Użytkownicy polskiego Logo na ATARI upowszechnionego na kasecie wydanej przez „Bajtka" mogą korzystać z instrukcji do angielskiej wersji tego je-zyka. Większość korzysta jednak z opisu polskiej wersji Sinclair LOGO. Aby udostępnić pełne możliwości ATARI LOGO, zamieszczamy słownik słów nie występujących w innych dialektach. W wykazie zastosowano skróty opisane w Słowniku LOGO w „Bajtku" nr 3-4/86.

1. GRAFIKA ŻÓŁWIA

GR

Przeznaczenie całej powierzchni ekranu na grafikę:wszystkie teksty są usuwane z ekranu (ale nie z pamięci).

TS

Przeznaczenie całej powierzchni ekranu na teksty;grafika jest usuwana z ekranu (ale nie z pamięci).

EP (Ekran Podzielony)

Dzieli powierzchnię ekranu pomiędzy grafikę i tekst;okno tekstowe znajduje się zawsze w dolnej części ekranu.

WZOR n

Powoduje nadanie żółwiowi kształtu określonego przez podany wzór. Wzór 0 |est standardowym kształtem żółwia, a wzory 1—15 są definiowane przez użytkownika.

WZOR^ <- n

Zwraca numer aktualnie używanego wzoru żółwia.

TEMPO n

Powoduje poruszanie się żółwi z podaną prędkością; żółwie można zatrzymać jedynie podaniem rozkazu TEMPO 0 lub CS. Wartość n musi być z zakresu od-199 do 199.

TEMPO^ <- n

Zwraca wartość prędkości żółwi.

KAZ n **KAZ l**

Powoduje, że wszystkie następne polecenia (oprócz PROS i KAŻDY) będą wykonywane przez żółwia o numerze „n" lub żółwie o numerach podanych w liście „l". Żółwie są numerowane od 0 do 3.

PROS n l **PROS l l**

Powoduje wykonanie poleceń z listy „l" przez określonego żółwia lub żółwie; użycie PROS nie wpływa na numery używanych żółwi określone poleceniem KAZ.

KTO <- n **KTO <- l**

Zwraca numer lub listę numerów aktualnie używanych żółwi (ustawionych instrukcją KAZ).

KAZDY l

Wszystkie żółwie KOLEJNO wykonują listę. Oto przykład użycia czterech powyższych poleceń:

```
KAZ [0 12 3]
KAŻDY [PRAWO 90 * KTO]
NAPRZOD 40
PROS [0 2] [WSTECZ 20]
```

KOLOR n

Zmienia kolor żółwia na podany jako parametr.

KOLOR^ <- n

Zwraca liczbę określającą aktualny kolor żółwia. Po starcie LOGO żółw 0 jest biały (7), żółw 1 pomarańczowy (20), żółw 2 purpurowy (44), a żółw 3 niebieski (68).

PISAK n

Powoduje używanie przez żółwia pisaka o podanym numerze; pisaki mają numery 0, 1 i 2.

PISAK^ <- n

Zwraca numer aktualnie używanego pisaka; w chwili startu LOGO wszystkie żółwie używają pisaka 0.

KP n n

Przydziela pisakowi o numerze podanym jako pierwszy parametr kolor o numerze podanym jako drugi parametr. Jednocześnie zmieniany jest kolor wszyskich linii już narysowanych tym pisakiem.

KP^ n <- n

Zwraca kod koloru pisaka o podanym numerze. W chwili startu pisak 0 ma kolor złoty (15), pisak 1 purpurowy (47), a pisak 2 pomarańczowy (121).

PP^ <- s

Wynikiem tej operacji jest słowo określające aktualne położenie pisaka: OPU, POD, ODWR lub ŚCIER.

1.1. KODY KOLORÓW

Interpreter LOGO ATARI dysponuje 16 barwami w 8 odcieniach każdy, a więc razem 128 kolorami, o następujących kodach:

0 -7	szary
8 -15	złoty
16 -23	pomarańczowy
24 - 31	czerwonopomarańczowy
32 - 39	różowy
40 - 47	purpurowy
48 - 55	purpurowoniebieski
56 - 63	niebieski
64 - 71	niebieski
72 - 79	jasnoniebieski
80 - 87	turkusowy
88 - 95	zielononiebieski
96 -103	zielony
104 -111	żółtozielony
112 -119	pomarańczowozielony
120 -127	jasnopomarańczowy

W każdym kolorze mniejsza liczba oznacza ciemniejszy odcień koloru, a większa |aśniejszy. np. 0—czarny, 7—biały.

2. DŹWIĘK

NUTA n n n n

Generuje dźwięk. Kolejne parametry określają: numer generatora (0 lub 1), częstotliwość dźwięku w jednostkach równych 5/6 Hz (częstotliwość napięcia sieciowego w USA wynosi 60 Hz, a w Polsce 50 Hz), głośność (0-15) i czas trwania w 1/50 sekundy. Następna instrukcja NUTA jest wykonywana dopiero po zakończeniu poprzedniej.

DZWIEK n n

Służy do ustalania brzmienia dźwięku. Pierwszy parametr określa numer generatora; a drugi wskazuje co ile jednostek czasu głośność ma być zmniejszana o 1 (podanie wartości 0 daje dźwięk o stałej głośności).

3.INSTRUKCJE OGÓLNE

GDY n l

Powołuje do życia Strażnika. Gdy kiedykolwiek w czasie aktywności Strażnika wystąpi warunek określony kodem „n", to jest wykonywana lista „l". Strażnika usuwa się poleceniem CS lub GDY n []. Kody warunków są podane poniżej.

Wejście żółwia na linię narysowaną pisakiem:

KOD	NR ŻÓŁWIA	NR PISAKA
0	0	0
1	0	1
2	0	2
3wciśnięcie przycisku drążka		
4	1	0
5	1	1
6	1	2
7 wyzwalany co sekundę		
8	2	0
9	2	1
10	2	2
11 nie używany		
12	3	0
13	3	1
14	3	2
15 zmiana położenia drążka		

Zderzenie między żółwiami

KOD	NR ŻÓŁWIA	NR ŻÓŁWIA
16	0	3
17	1	3
18	2	3
19	1	0
20	2	0
21	2	1

WARUNEK n <- s

Zwraca wartość PRAWDA, jeśli w chwili wykonywania warunek o numerze n jest spełniony i wartość FAŁSZ, jeśli nie jest spełniony.

WEJSCIE n n <- n

Zwraca kod określający wejście przez żółwia o numerze podanym jako pierwszy parametr na linię narysowaną pisakiem o numerze podanym jako drugi parametr (zob. GDY i WARUNEK).

KOLIZJA n n <- n

Zwraca kod określający zderzenie żółwi o podanych numerach (zob. GDY i WARUNEK).

4. INSTRUKCJE ARYTMETYCZNE

POWTLOS

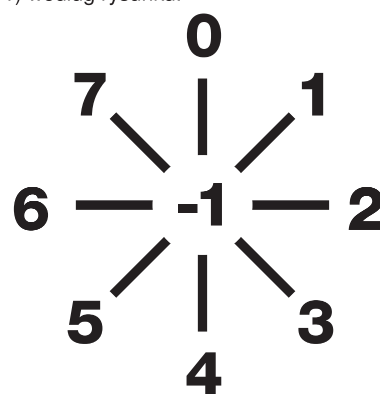
Zapamiętuje kolejne wartości funkcji LOSOWA i pozwala na ich odtworzenie (parametr funkcji LOSOWA musi być taki sam).

```
POWTORZ 5 [WPISZ LOSOWA 10] PS
POWTLOS POWTORZ 5 [WPISZ LOSOWA 10] PS
POWTLOS POWTORZ 5 [WPISZ LOSOWA 10] PS
POWTLOS POWTORZ 5 [WPISZ LOSOWA 7] PS
```

5. INSTRUKCJE MANIPULATORÓW

DR n <- n

Zwraca liczbę określającą położenie podanego drążka (0 lub 1) według rysunku.



WIOS n <- n

Zwraca liczbę z przedziału od 0 do 247 określającą położenie podanego wiosełka (0-3).

PDR n <- s

Jeżeli przycisk podanego drążka jest wciśnięty, to zwraca wartość PRAWDA; w przeciwnym wypadku FAŁSZ.

PWIOS n <- s

Jeżeli przycisk podanego wiosełka jest wciśnięty, to zwraca wartość PRAWDA: w przeciwnym wypadku FAŁSZ.

6. INSTRUKCJE REDAKCYJNE

REDWZ n

Powoduje przejście do trybu redagowania wzoru, wyświetlając siatkę wzoru o podanym numerze. Kursor przesuwa się po siatce wzoru przy użyciu klawiszy ze strzałkami i klawisza CONTROL. Wypełnienie lub opróż nenie pola siatki następuje po naciśnięciu spacji. Tryb redagowania opuszcza się naciskając klawisz ESC (nowy wzór zostaje zapamiętany) lub BREAK (w pamięci pozostaje poprzedni wzór).

DAJWZ n <- l
Zwraca listę 16 liczb określających wartości bajtów podanego wzoru.
WEZWZ n l
Definiuje podany wzór wedtug listy 16 liczba (liczby muszą być z zakresu 0-255).

WEZWZ 1 [255 129 129 129 129 129 129 129 129 129 129 129 129 129 129 255]
CS WZOR 1
PISZ DAJWZ 1

POS n **POS l**
Drukuje treść podanego Strażnika lub Strażników.
POWS
Drukuje treść wszystkich aktualnie czynnych Strażników.
7. INSTRUKCJE WEJŚCIA/WYJŚCIA
LADUJ S
Wczytuje procedury i zmienne z podanego urządzenia zewnętrznego. Dla odczytania z magnetofonu należy użyć:
LADUJ"C:

ZAPISZ s
Zapisuje procedury i zmienne na podanym urządzeniu zewnętrznym. W celu zapisania własnych procedur na taśmie należy użyć: ZAPISZ „C:

USPL s
Usuwa plik o podanej nazwie z urządzenia zewnętrznego. Nie działa z magnetofonem.
KATALOG s
Podaje spis plików zawartych w urządzeniu zewnętrznym. Nie działa z magnetofonem.

ZAPIS s **ZAPIS [ ]**
Otwiera urządzenie zewnętrzne do zapisu danych (zapisywane są wszystkie znaki ukazujące się na ekranie). Odpowiada instrukcji BASIC-u OPEN # 1,8,0,s. Dla magnetofonu:
ZAPIS „C:
a dla drukarki:
ZAPIS „P:
Zamknięcie urządzenia następuje po poleceniu ZAPIS [].

ODCZYT s **ODCZYT[ ]**
Otwiera urządzenie zewnętrzne, z którego będą odczytywane dane przez instrukcje CZ i CL. Odpowiada instrukcji BASIC-u OPEN # 1,4,0,s. Dla magnetofonu:
ODCZYT"C:
Urządzenie zamyka się instrukcją ODCZYT [].

8. PRACA KLAWIATURY
Oprócz normalnych klawiszy funkcyjnych w ATARI LOGO są przewidziane pewne kombinacje klawiszy spełniające dodatkowe funkcje:

CTRL + A — przesunięcie kursora na początek bieżącej linii;

CTRL + E — przesunięcie kursora na koniec bieżącej linii;

CTRL + Y — wyświetlenie ostatnio wpisanej linii. Ponadto w EDYTORZE (w trybie redagowania) działają następujące klawisze:

CTRL + X — przesunięcie kursora na początek redagowanego tekstu;

CTRL + Z — przesunięcie kursora na koniec redagowanego tekstu;

CTRL + V — przesunięcie obrazu do następnej strony edytora;

CTRL + W — przesunięcie obrazu do poprzednie] strony edytora;

CTRL + CLEAR — usuwa tekst w prawo od kursora do końca linii i umieszcza go w buforze usuwania (do 110 znaków);

SHIFT + DELETE — jak CTRL + CLEAR;

CTRL + Y — wypisuje zawartość bufora usuwania; RETURN — tworzy nową linię i umieszcza kursor na jej początku;

CTRL + INSERT — tworzy nową linię, lecz kursor pozostaje na końcu starej;

SHIFT + INSERT — jak CTRL + INSERT;

ESC — opuszczenie edytora z zapamiętaniem wprowadzonych procedur i zmian;

BREAK — opuszczenie edytora; wszystkie poprawki i nowo wprowadzone procedury zostają zapomniane.

Wojciech Zientara

ZNAKI GRAFICZNE W LOGO

Poniższy zestaw procedur może przydać sie użytkownikom LOGO na komputerze Spectrum — zarówno angielskiej, jak i (po przetłumaczeniu słów specjalnych) polskiej wersji tego języka. Umożliwia on stosunkowo wygodne definiowanie znaków graficznych użytkownika, dostępnych na Spectrum.

Jądro programu stanowią trzy procedury, mogące znaleźć zastosowanie w innych programach w LOGO.

```
TO GPAF  :Dl'ZA.LITEP.A
OP CHAR 79 + ASCII   :DUZA.LITERA
END

TO SCHOWAJ  :ADPFS  :CIAG.BAJTÓW IF
EMPTYP   :CIAG.BAJTÓW   [STOP] .DEPOS-
IT  :ADRES FIRST  :CIĄG.BAJTÓW SCHOWAJ
:ADREE +   ]  BF  :CIAG.BAJTÓW END

TO USR  : DUZA. LITERA
OP 65368 ♦ 8 „  ( -65 + ASCII   :DUZA.
LITERA   )
END
```

Operacja GRAF umożliwia dostęp do danego znaku (oznaczonego przy pomocy dużej litery) w trybie graficznym. Wszystkich takich znaków jest 21; oznaczać je w LOGO możemy teraz GRAF „A ... GRAF"U.

Procedura SCHOWAJ umożliwia umieszczenie wskazanego ciągu bajtów w kolejnych komórkach pamięci, począwszy od adresu podanego jako pierwszy parametr. Z kolei wynikiem operacji USR jest, podobnie jak w BASIC-u, adres komórki pamięci, od której zaczyna się ośmiobajtowy wzorzec znaku graficznego. Tutaj także parametr musi być dużą literą z przedziału „A ... „U. Oto przykład zastosowania tych procedur.

```
SHOWAJ USR „F  [170 65 85 170
170 65 85 170]
PRINT GRAF  „F
```

Proces definiowania znaków graficznych możemy usprawnić, budując prosty edytor tych znaków. W tym celu zdefiniujemy dalsze procedury:

```
TO BIN  :ZAPIS
IF EMPTYP  :ZAPIS   [OP 0]
IF EQUALP „# LAST :ZAPIS [MAKE
„BIT 1] [MAKE  „BIT 0]
OP   :BIT +  ?      BIN BL
;ZAFIS END
```

Operacja BIN przelicza liczbę w układzie dwójkowym zapisaną „graficznie" — przy pomocy ciągu znaków #=1 i .=0 na wartość dziesiętną.

```
PRINT BIN „###..#. 114
```

Procedura DEFINIUJ jest natomiast właściwym edytorem.

```
TO  DEFINIUJ   :DUZA.LITFRA PRINT   „
PRINT GRAF   :DUZA.LITERA PRINT   „
MAKE  „LISTA.DEF   [I MAKE   „WERS   0
REPEAT 8   [TYPE WORD  :WERS  „:  MAKE
„LISTA.DEF
LPUT BIN FIRST RL  :LISTA.DEF MAKE „WERS
:WERS +  1]
PRINT „
PRINT  [O. K.   ?]
IF MFMBERP RC [N n] [DEFINIUJ :DUZA.
LITERA STOP] PRINT „
SCHOWAJ USR :DUZA.LITERA :LISTA.DEF PRINT
„
PRINT GRAF :DUZA.LITEPA
PRINT  „
END
```

Procedurą DEFINIUJ posłużymy się np. w następujący sposób:

```
DEFINIUJ „A
A
0   ...###..
1   ..#.....
2   ...##...
3   ...#.#..
4   ....##..
5   ......#.
6   ...###..
7   ........
0.  K?
§
```

Na zakończenie zapoznajmy się jeszcze z dwiema procedurami umożliwiającymi nagranie znaków graficznych na taśmie i ich powtórne wczytanie do pamięci.

```
TO NAGRAJ.GRAF  :NAZWA
.BSAVE :NAZWA [65368 168]
END
TO LADUJ.GRAF :NAZWA .BLOAD
:NAZWA  65368 END
```

Marcin Waligórski

DUSZKI

Wszystkie opisy komputerów Atari wspominają, że jest możliwe uzyskanie na ekranie „duszków” (sprite). Niestety poza ogólnikowymi lub niedokładnymi wiadomościami nigdzie nie można znaleźć przystępnej informacji, jak je otrzymać.

Aby nasze rozważania były zrozumiałe dla wszystkich, trzeba rozpocząć od zrozumienia pojęcia bajtu. Bajt jest złożony z ośmiu bitów. Bity w bajcie są numerowane według ważności od 0 do 7. W celu otrzymania wartości bajtu należy dodać do siebie wartości wszystkich bitów w tym bajcie pomnożonych przez 2 podniesione do potęgi równej numerowi bitu.

7	6	5	4	3	2	1	0

Np. 01101011 =	$0*2^7$	=	0*128	=	0
	$1*2^6$	=	1*64	=	64
	$1*2^5$	=	1*32	=	32
	$0*2^4$	=	0*16	=	0
	$1*2^3$	=	1*8	=	8
	$0*2^2$	=	0*4	=	0
	$1*2^1$	=	1*2	=	2
+	$1*2^0$	=	1*1	=	1
					107

Zrozumiałe jest teraz, dlaczego największą możliwą wartością bajtu jest 255 (proszę sprawdzić, że bin 11111111 = dec 255). Pozostaje nam wyjaśnienie pojęcia „rozdzielczość”. Otóż oznacza ono w tym przypadku ile linii ekranu będzie zajęte przez jedną linię duszka. Przy rozdzielczości jednowierszowej każdej linii ekranu odpowiada jedna linia duszka, a przy dwuwierszowej jedna linia duszka zajmuje dwie linie ekranu. Przykład duszka w różnych stopniach rozdzielczości jest pokazany na rysunku 1.

Wiedząc jeszcze, że szerokość duszka w Atari wynosi 1 bajt, a maksymalna wysokość 128 lub 256 bajtów (zależnie od rozdzielczości) możemy przystąpić do pierwszego etapu pracy, jakim jest...

PROJEKTOWANIE DUSZKA

Najpierw trzeba narysować siatkę wyobrażającą bity naszego duszka, oczywiście uwzględniając przy tym założoną rozdzielczość. Następnie wypełniamy siatkę wzorem i obliczamy wartości poszczególnych bajtów. Przykład takiego wzoru i obliczenia jest na rysunku 2. Teraz należy znaleźć dla duszka bezpieczne...

MIEJSCE W PAMIĘCI

Wielkość miejsca w pamięci potrzebnego dla naszego duszka (zwanego oficjalnie grafiką graczy i pocisków — Player/Missile Graphics — P/MG) jest zależna od zastosowanej rozdzielczości i wynosi 2 KB dla rozdzielczości jednowierszowej lub 1 KB dla dwuwierszowej. Oprócz tego blok pamięci P/MG musi rozpoczynać się odpowiednio na granicy pełnych 2 KB lub 1 KB. W jaki sposób znaleźć taki obszar pamięci zabezpieczony przed ingerencją BASIC-u i systemu operacyjnego? Jest to dość trudne. Nie będziemy więc szukać, lecz sami STWORZYMY taki obszar.

W chwili uruchomienia komputera system operacyjny sprawdza, ile pamięci RAM ma do dyspozycji i wielkość tej pamięci wyrażoną w stronach (1 strona = 256 bajtów) umieszcza w komórce pamięci zwanej RAM-TOP pod adresem 106. Oszukamy więc komputer wpisując do tej komórki odpowiednio mniejszą wartość.

POKE 106,PEEK(106)-8 dla rozdzielczości jednowierszowej (8*256=2048)

POKE 106,PEEK(106)-4 dla rozdzielczości dwuwierszowej (4*256 = 1024)

Jeżeli po wykonaniu tej komendy wpiszemy coś do chronionego obszaru, to pojawią się na ekranie jakieś „śmieci”. Powód jest prosty. System operacyjny wie już, że pamięć jest mniejsza, ale nie wie o tym ANTIC — mikroprocesor tworzący obraz. Aby system operacyjny „przekazał” mu tę informację wystarczy polecić wykonanie dowolnej instrukcji GRAPHICS. Czy można już wpisać dane duszka do utworzonego obszaru pamięci? Nie znamy jeszcze dokładnego miejsca. W tym celu musimy zapoznać się z mapą pamięci P/MG (rys. 3)

Jak widzimy w obszarze pamięci P/MG dysponujemy polami po 128 lub 256 bajtów dla każdego gracza i polem 128 lub 256 bajtów dla czterech pocisków. Na ekranie każde takie pole będzie miało wygląd pionowego pasa o szerokości ośmiu punktów (odpowiadających bitom wzoru). Od tego, gdzie w obrębie pola umieścimy nasze dane, będzie zależało pionowe położenie duszka na ekranie. Załóżmy, że rysujemy gracza 0 i chcemy umieścić go mniej więcej w środku ekranu. Proszę wpisać i uruchomić poniższy program.

```
10 POKE 106,PEEK(106)-9:GRAPHICS
0:PMB=PEEK(186):REM PMB -  POC-
ZATEK PAMIECI P/MG
20 FOR I=PMB*256+1024 TO
PMB*256+2048:POKE I,0:NEXT I:REM
KASOWANIE PAMIECI P/MG
30 FOR I = 1 TO 21 : READ A:
POKE PMB*256+1145+I,A:NEXT I:REM
ODCZYT DANYCH DUSZKA
40 DATA 12,24,56,124,34,124,40,5
6,56,60,126,251,185,56,56,60,28,
28,14,6,3
```

I co? Nic. To oczywiste. Z tego, że my wiemy gdzie są dane duszka, nie wynika, że komputer też o tym wie. Trzeba mu powiedzieć, gdzie i jakie duszki umieściliśmy w pamięci oraz gdzie chcemy je zobaczyć na ekranie. Jest na to tylko jeden sposób:

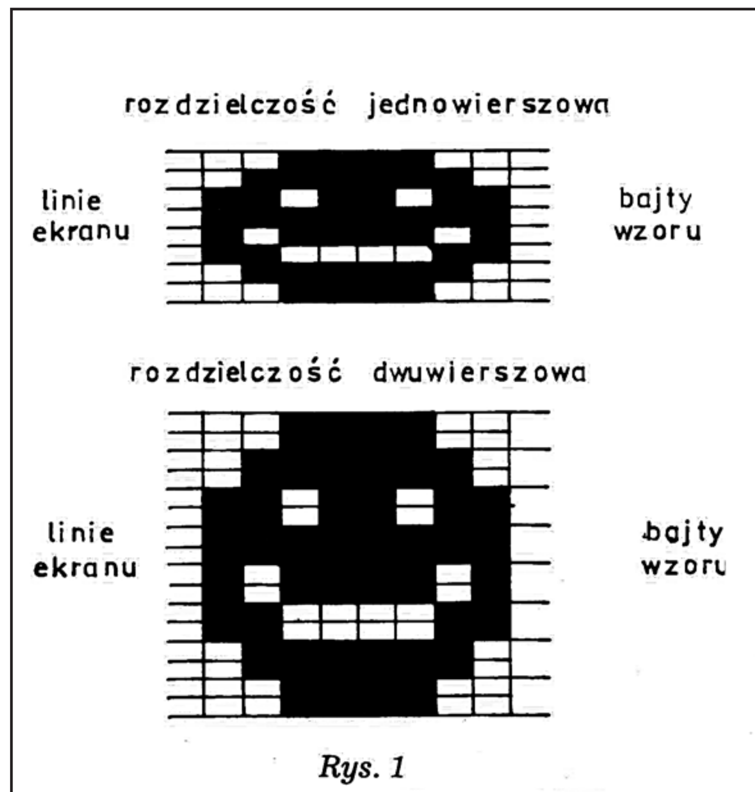
POKE,POKE,POKE...

W tym miejscu muszę podać znaczenie i adresy kilkunastu komórek pamięci Atari.

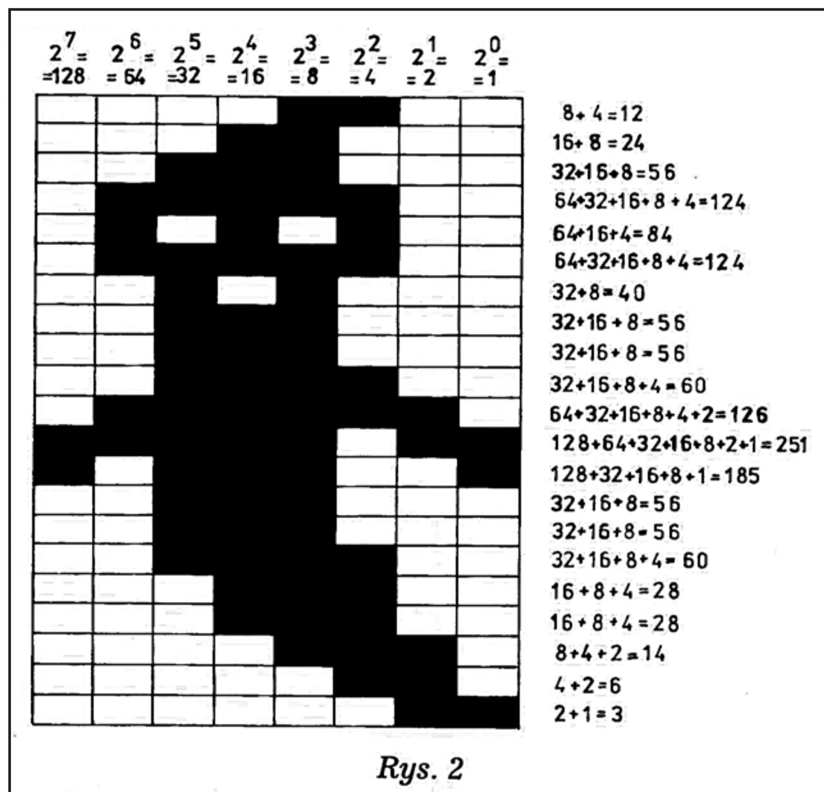
DMACNTL (559, Direct Memory Access CoNTroL — kontroler bezpośredniego dostępu do pamięci): bity 0—1 sterują szerokością obrazu (00 — brak, 01 — wąski, 10 — normalny, 11 — szeroki), bit 2 włącza DMA dla pocisków, bit 3 włącza DMA dla graczy, bit 4 służy do wyboru rozdzielczości P/MG (0 — dwuwierszowa, 1 — jednowierszowa), bit 5 włącza DMA dla programu ANTICa, a bity 6—7 są niewykorzystane.

GTIACNTL (623, GTIA CoNTroL — kontroler układu GTIA): bity 0—3 służą do wyboru priorytetu P/MG i tła. bit 4 łączy pociski w piątego gracza (kolor z COLPF3). bit 5 łączy parami graczy 0 i 1 oraz 2 i 3 (uzyskujemy graczy wielokolorowych), a bity 6—7 pozwalają otrzymać tryby GRAPHICS 9, 10 i 11. Priorytety duszków pola gry są podane w tabeli.

ustawiony bit	priorytety kolorów (kolory wcześniej wymienione zasłaniają kolory następne)
0	P/M0, P/M1, P/M2, P/M3, PF0, PF1, PF2, PF3, BAK

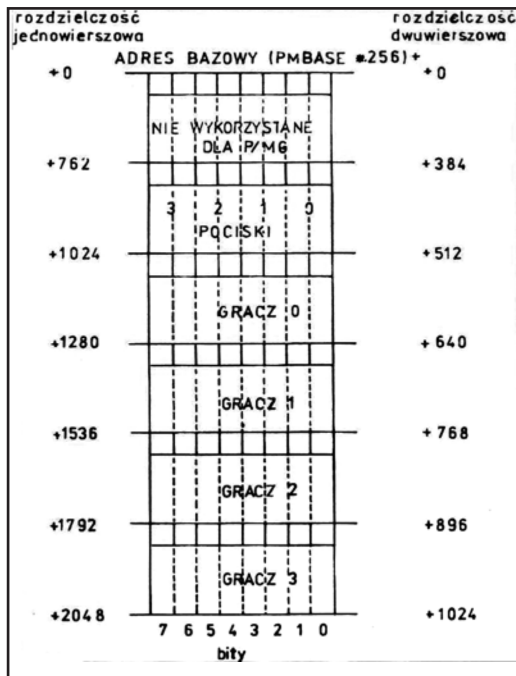


Rys. 1



Rys. 2

NIE BÓJ SIĘ PRZERWAŃ (3)



0 P/M0, P/M1, PF0, PF1, PF2, PF3, P/M2, P/M3, BAK

0 PF0, PF1, PF2, PF3, P/M0, P/M1, P/M2, P/M3, BAK

0 PF0, PF1, P/M0, P/M1, P/M2, P/M3, PF2, PF3, BAK

Uwaga: ustawienie kilku bitów spowoduje nieprzewidziane efekty.

PMCNTL (53277, Player/Missile CoNTroL — kontroler graczy i pocisków): bit 0 włącza pociski, bit 1 włącza graczy, bit 3 blokuje przyciski joysticków, pozostałe bity nie są wykorzystane.

PMBASE (54279, Player/Missile BASE — adres bazowy pamięci graczy i pocisków): zawiera starszy bajt adresu (numer strony) obszaru pamięci przeznaczonego dla P/MG.

SIZEPO-3, (53256-53259, SIZE Player — rozmiar gracza): bity 0—1 sterują wielkością poziomą gracza, pozostałe są niewykorzystane.

SIZEM (53260, SIZE Missiles — rozmiar pocisków): pary bitów 0—1, 2—3, 4—5 i 6—7 sterują odpowiednio poziomą wielkością pocisków 0—3.

HPOSPO-3, HPSMO-3 (53248-53255, Horizontal POSition Player/Missile — pozioma pozycja gracza/pocisku): w tych rejestrach wpisuje się pozycję poziomą graczy/pocisków.

COLPMO-3 (704-707, COLor of Player/Missile — kolor gracza i pocisku), **COLPFO-3** (708-711, COLor of PlayField — kolor pola gry), **COLBAK** (712, COLor of BAcKground — kolor tła): znaczenie tych komórek jest powszechnie znane — zawierają one wartości kolorów.

Po tej długiej liście czas na drugą część programu:

```
50 POKE 54273,PME:REM POCZATEK
PAMIECI P/MG
60 POKE 53277,2: REM bin
00000010 - WLACZENIE GRACZY
70 POKE 623,1:REM bin 00000001 -
USTAWIENIE PRIORYTETU
80 POKE 53256,0:REM NORMALNA
WIELKOSC GRACZA 0
90 POKE 53248,122:REM POZIOMA
POZYCJA GRACZA 0
100 POKE 704,14:REM KOLOR GRACZA
0 - BIALY
110 POKE 553,58:REM bin 00111010
- OBRAZ NORMALNY, DMA DLA
GRACZY, ROZDZIELCZOSC JEDNOWI-
ERSZOWA, DMA DLA OBRAZU
```

Uruchamiając cały program uzyskamy na środku ekranu zaprojektowanego duszka. Proponuję nieco z nim poeksperymentować w celu lepszego zrozumienia podanych zasad.

TO JUŻ KONIEC? PRZECIEŻ ON JEST NIERUCHOMY! Proszę się nie obawiać. O tym jak poruszać duszka napiszę w następnym numerze „Bajtka".

Wojciech Zientara

W pierwszym odcinku opisane zostało przerwanie wywoływane przez program ANTIC-a. Z tworzeniem obrazu jest związane jeszcze jedno przer-wanie, znacznie ciekawsze i dające więcej możliwości. Jest to przerwa-nie synchronizacji pionowej, zwane także pustym przerwaniem pionowym (Vertical BLanK Interrupt)..

Podczas tworzenia obrazu na ekranie monitora po osiągnięciu prawego, dolnego narożnika wiązka elektronów jest wygaszana i przesuwana do lewego, górnego narożnika. W tym czasie wykonywane jest przez system operacyjny przerwanie VBLK. Na jego końcu użytkownik może dołączyć swoją własną procedurę przerwania o długości do 2 KB!

KROK 1 (linie 10-20)

Piszemy przykładową procedurę, która co jedną sekundę będzie zmieniała kolor tła i umieszczamy ją w pamięci.

```
198 206       DEC 206
165 206       LDA 206
208  16       BNE END
173 200   2   LDA COLBAK§
 24           CLC
105   2       ADC #2
141 200   2   STA COLBAK§
141  26 208   STA COLBAK
169  50       LDA #50
133 206       STA 206
 76  98 228   JMP EXITVBLK
END
```

Procedura najpierw zmniejsza o 1 zawartość komórki 206 i ładuje ją do akumulatora. Jeśli jest ona różna od zera, to następuje skok na koniec procedury. Ponieważ przerwanie VBLK jest wykonywane co 1/50 sekundy, powodujemy w ten sposób, że nasza procedura jest wykonywana tylko raz na 50 przerwań, a więc co jedną sekundę. Następnie zawartość rejestru koloru tła jest ładowana do akumulatora, zwiększana o 2 i ponownie zapisywana do rejestru koloru. Dla uniknięcia niespodzianek wpisujemy ją zarówno do rejestru sprzętowego (COLBAK); jak i do rejestru-cienia (COLBAK§). Na zakończenie należy jeszcze odtworzyć wartość opóźnienia (50) w komórce 206. Teraz spodziewalibyśmy się instrukcji RTI — powrót z przerwania. Ponieważ jednak system operacyjny musi wykonać jeszcze kilka operacji, wykonujemy w tym miejscu skok do EXITVBLK (wyjście z VBLK).

KROK 2 (linie 30-40)

Procedura przerwania VBLK wymaga specjalnego sposobu uruchomienia. Wykonuje się to przy pomocy krótkiej pro-cedurki, którą także musimy wpisać do pamięci.

```
104        PLA
160   0    LDY #0
162   6    LDX #6
169   7    LDA #7 76
 92  228 JMP SETVBLK
```

Do rejestru Y wpisujemy młodszy bajt adresu naszej procedury przerwania, do rejestru X starszy bajt (0 + 256*6=1536), a do akumulatora kod zmienianej procedury (7 — przerwanie użytkownika po przerwaniu VBLK). Następnie wykonujemy skok do procedury SETVBLK (ustawianie VBLK). Początkowa instrukcja PLA jest konieczna ze względu na sposób działania instrukcji BASIC-u USR. KROK 3 (linia 50)

Pozostało tylko wpisanie wartości początkowej do komórki 206, wywołanie procedury ustawiającej naszą procedurę przerwania i zezwolenia na przerwanie VBLK (bit 6 w rejestrze NMIEN — 54286).

Oto cały program w BASIC-u:

(ziew)

```
10 GRAPHICS 0:FOR I=1536 TO 1560:READ A:POKE
I,A:NEXT I
20 DATA 198,206,165,206,208,16,173,200,2,24
,105,2,141,200,2,141,26,208,169,50,133,206,
76,98,228
30 FOR I=1561 TO 1570:READ A:POKE I,A:NEXT I
40 DATA 104,160,0,162,6,169,7,76,92,228
50 POKE 206,50:I=USR(1561):POKE 54286,64
```

KLAN ATARI ST

Klan Atari ST powstał 1 sierpnia 1986 roku. Posiada obecnie 10 członków. Celem klanu jest popularyzacja ATARI serii ST, jak również wymiana doświadczeń, literatury oraz oprogramowania.

Klan jest przeciwnikiem bezkrytycznego szaleństwa gier komputerowych, choć nie przeczymy, że gry są wspaniałą rozrywką w wolnych chwilach. Jesteśmy także przeciwnikami zbijania fortuny, na sprzedawaniu nielegalnie skopiowanych programów. Wychodząc na przeciw tym założeniom, założyliśmy bank programów użytkowych.

Członkiem klanu może zostać każdy, kto zgadza się z powyższymi twierdzeniami oraz prześle swoie zgłoszenie na nasz adres: Należy podać imię, nazwisko, dokładny adres, w miarę możliwości telefon, typ komputera wraz z konfiguracją oraz zainteresowanie konkretnym wykorzystaniem ATARI ST.

Bank programów posiada obecnie około 40 pozycji w tym: języki programowania (PASCAL, C, ASSEMBLER, FORTH i inne), CPM-280, edytory tekstu (1-ST WORD. EDIT), bazy danych (DATA BASE MASTER I),programy graficzne (NEO, DR. DOOPLE), rozrywkowe programy strategiczne, polsko-języczne opisy tych programów oraz inne. Każdy członek klanu, może otrzymywać je bezpłatnie po spełnieniu następujących warunków:

1) należy odstąpić klanowi 1 dyskietkę, która zostanie włączona do banku programów (nieodpłatnie),

2) w miarę możliwości dostarczać nowe programy,

3) pokryć koszty przesyłki zwrotnej.

Klan posiada także bibliotekę literatury. W chwili obecnej jest do dyspozycji 6 książek. Kopie można otrzymać po opłaceniu kosztów wykonania odbitek kserograficznych, lub przez wymianę na inną pozycję. Klan wysyła na życzenie spis członków, literatury oraz programów, po dostarczeniu znaczków za 25 złotych.

Od 1987 roku ukazywał się będzie biuletyn KLANU ATARI ST. Każdy, kto chciałby zostać współredaktorem biuletynu, proszony jest o przesłanie artykułów na adres Klanu. Posiadamy także kontakty z użytkownikami ATARI ST na zachodzie. Pozwala to na dostęp do nowych programów w miarę szybkim czasie. Wszelkie dalsze kontakty będą mile widziane.

Klan nie bierze odpowiedzialności za prywatne postępowanie członków, choć w wypadku złamania statutu wyciąga konsekwencje, do skreślenia z Klanu włącznie.

Prosimy także o nadsyłanie słów krytyki, uwag i propozycji. Przyczynią się one z pewnością do rozszerzenia działalności Klanu.

Zarząd KLANU ATARI ST

Adres: 42-600 Tarnowskie Góry
ul Narutowicza 3

POLSKIE ZNAKI (2)

Spieszę podać parę uzupełnień do programu pozwalającego otrzymać polskie litery na ekranie ATARI, zamieszczonego w BAJTKU nr 2/87. Mam nadzieję, że zachowaliście na ka-secie lub dyskietce ów program — będzie Wam dziś potrzebny!

I. REDEFINICJA KLAWISZY

Zacznę od uzasadnienia mojego przyporządkowania klawiszy polskim literom. Otóż chciałam, aby nowy zestaw znaków mógł być z powodzeniem używany w tekstowych trybach graficznych 0, 1 i 2. W trybie 0 (GRAPHICS 0) Atari używa całego zestawu naraz, ale w trybach 1 i 2 dostępna jest w danej chwili tylko połowa znaków!

W instrukcji swego Atari możecie znaleźć spis kodów ATASCII wszystkich 128 znaków. Pierwsza ćwiartka (kody 0-31) zawiera znaki graficzne, druga (32-63) — cyfry i większość symboli, trzecia (64-95) — głównie wielkie litery, czwarta (96-127) — małe. Jest kilka odstępstw — wykryjcie je sami.

W zestawie, z którego Atari korzysta przy obsłudze ekranu, kolejność jest nieco inna. Pierwsza ćwiartka staje się trzecią, druga i trzecia — odpowiednio pierwszą i drugą; czwarta pozostaje na miejscu. Tak właśnie ułożona jest mapa bitowa wszystkich znaków, zajmująca dokładnie 1 KB (128 znaków, 8 bajtów na znak) i zaczynająca się — po naszej modyfikacji — pod adresem 156 * 256 = 39936. W trybach 1 i 2 po komendzie POKE 756, 156 Atari „widzi” ćwiartkę pierwszą i drugą, czyli duże litery, cyfry i symbole. Traci je z oczu, gdy rozkażesz POKE 756, 158: teraz uznaje małe litery i znaki graficzne. Oto dlaczego dużym polskim literom przyporządkowałem sekwencje klawiszy z SHIFT, odpowiadające rzadziej używanym symbolom, zaś małe litery znalazły się w zestawie zamiast niektórych znaków graficznych. W przeciwnym razie nie mógłbyś napisać „bąk”, gdyż w połówce zestawu zawierającej małe „b” nie byłoby małego „ą”. Nie pomogłoby użycie CAPS ani „Inverse Video”, gdyż w trybach 1 i 2 powoduje tylko zmianę koloru znaku.

Zamieszczony dziś program pozwala jednak na dowolną zmianę przyporządkowania klawiszy. Ostrzega przed skutkami „złego” wyboru, lecz nie wymusza jego zmiany. Oto sposób użycia:

a) wczytaj z kasety lub dyskietki program z Bajtka nr 2/87;
b) wpisz zamieszczony obok program nr 1;
c) napisz RUN. Po 30 sekundach program zapyta, jak chcesz otrzymywać małą literę „ą”. Wcześniej mrugnie kilka razy, przypominając obecne przyporządkowanie;

— jeśli nie chcesz go zmieniać, naciśnij RETURN, po czym odpowiedz „Nie” na pytanie, czy jednak chcesz je zmienić. Zostaniesz natychmiast zapytany o następną literę;

— jeśli chcesz je zmienić, naciśnij klawisze odpowiadające Twemu wyborowi. Nie zapomnij o CON-TROL (aby uzyskać np. znak graficzny) lub CAPS (dla małej litery) itd;

— otrzymawszy informację o wszystkich 28 znakach (9 małych, 9 dużych), program ruszy dalej. Po każdej literze program może Cię ostrzec:

a) „Tylko Gr. 0!” — jeśli zignorujesz to ostrzeżenie, w trybach 1 i 2 dana litera znajdzie się w obcym sobie otoczeniu (mała wśród dużych lub duża wśród małych);
b) „Tylko BOOT!” — możesz zignorować, jeśli zamierzasz utworzyć „boot-cassette” lub „boot-disk” (patrz dalej). Rzecz dotyczy tylko dwóch sekwencji klawiszy: małego „s” oraz „CONTROL-D”;
c) „2 razy!” — jeśli je zignorujesz, tej polskiej litery po prostu nie będzie. Pozwala Ci to utworzyć ograniczony zestaw, np. zawierający tylko małe polskie litery, za to o 9 więcej oryginalnych znaków Atari. Jeśli więc chcesz zrezygnować z jakiejś polskiej litery, wprowadź sekwencję klawiszy, której już raz użyłeś.

Jak widzisz, redefinicja klawiszy może być prawie dowolna. Jeśli moje wyjaśnienia wydadzą Ci się niepełne — poeksperymentuj.

II. BOOT-CASSETTE I BOOT-DISK

Te jeszcze dziś nie dające się przełożyć na polski sformułowania oznaczają kasetę lub dyskietkę zawierającą program, który wykonuje się natychmiast po włączeniu komputera. W poprzednim odcinku obiecywałem „nierelokowalną” wersję mojego programu tylko posiadaczom stacji dysków. Później jednak uznałem, że magnetofon nie zasługuje na taką dyskryminację. Zamiast napisanych w BASIC-u procedur, dołączanych do każdego programu, możemy uzyskać polskie litery dzięki procedurze maszynowej urzędującej na stronie 6 pamięci (1536-1791, hex 600-6FF). BASIC nie rusza strony 6, nie szkodzi jej także naciśnięcie RE-SET. Jak to zrobić? Ano tak:

a) wczytaj program z Bajtka 2/87;
b) wpisz do pamięci programy 2 i 3;
c) napisz RUN i poczekaj na pytanie „Kaseta czy Dysk?”;

— odpowiedz „K” + RETURN, przewiń kasetę, wciśnij RECORD i PLAY, uderz RETURN jeszcze dwakroć, albo

— odpowiedz „D” + RETURN, włóż dyskietkę z DOSem (FMS.SYS w wypadku DOSa 3, DOS.SYS i DUP.SYS dla DOSa 2.0 i 2.5), naciśnij RETURN jeszcze raz; dyskietka z DOS 2.0 lub 2.5 musi zawierać co najmniej 4 wolne sektory, z DOS 3 — 1 blok;
d) i gotowe!

Jak tego używać? Dyskietkę wysłarczy włożyć do stacji i włączyć komputer; w wypadku magnetofonu włączaj Atari z wciśniętym klawiszem START, przewiń kasetę we właściwe miejsce. Jeśli nowy zestaw znaków zniknie z pamięci (np. po naciśnięciu Reset), przywróć go sekwencją POKE 106,156:GRAPHICS 0:X = USR (1698):POKE 756,156.

Pamiętaj, aby po każdej komendzie GRAPHICS powtórzyć POKE 756,156, albo — w trybach 1 i 2

— POKE 756,158 w celu dostępu do małych liter etc.

Aby w trybach 1 i 2 obok małych liter uzyskać spację, rozkaż X = USR (1763). Procedura ta przy

```
*** Listing 1 ***

200 GOSUB 32764:POKE 82,0:OPEN
#1,4,0,"K:":C=32:X=756:FOR I=1 TO 18:D=9*I
210 L=ASC(POL$(D,D)):A=INT(L/8):R=L-
8*A:B=A+C*(R+(R<2)-2*(R=2)):? CHR$(125):?
220 ? "Podaj klawisz dla ";CHR$(B);" = ";:B=L:FOR
J=0 TO 6:FOR K=0 TO 99:NEXT K
230 POKE X,224:FOR K=0 TO 99:NEXT K:POKE X,156:NEXT
J:GET #1,A:? CHR$(A):K=0
240 ON A=155 GOTO 270:B=A-128*(A>127):B=A-
128*(A>127):B=B+64*(B<C)-C*(B>31)*(B<96):M=INT(B/C)
250 B=8*B-255*M:A$="2 razy!":FOR J=1 TO I-1:K=K+(ASC
(POL$(9*J,9*J))=B):NEXT J
260 ON K$I=K>0 GOSUB 980:A$="Tylko BOOT!":ON
(B=34)+(B=155) GOSUB 980
270 A$="OK.":GOSUB 980:POL$(D,D)=CHR$(B):NEXT
I:CLOSE #1
980 ? :? :? A$;" Zmienisz? (T/N) ";
981 GET #1,A:A=A-C*(A>90):N=(A=78):ON 1-N-(A=84)
GOTO 981:IF N THE RETURN
982 POP: GOTO 210
```

```
*** Listing  2 ***

400 ? CHR$(125):POL$(223,223)=CHR$(6):POL$(241,241)=
CHR$(6): POL$(195)=POL$(218);DIM AUT$(250),
B$(99):RESTORE 405:M=0
405 DATA 000400800D80A93C8D02D31860A2FDB
DBA809DFF05CAD0F7A200BD1A03C945F005E8E8E
8D0F4E88E8180BD1A0385CD,405,261447
410 DATA A9839D1A03E8BD1A0385CEA9809D1A0
3A000A210B1CD99B380C8CAD0F7A95BBD87B0A9B
08D8880A9288D828060AC82,410,254911
415 DATA 80F009B99280CE280A00160B8A48AE8
180A5CD9D1A03E8A5CE9D1A0368AAA99BA001600
0000000000000000000000,415,237765
420 DATA 00004C0000003635312C363537454B4
F503A2938393631285253553D593A302E52473A3
430 FOR K=1 TO 4:READ B$,L,S1:S2=0:for J
=1 TO 93-2*(K=4) STEP 2:B=C:FOR I=J TO J
+1:N=ASC(B$(I,I)):S2=S2+I*N
440 N=N-48-7*(N>57):B=B+N*(1+15*(I=J)):N
EXT i:M=M+1:AUT$(M)=CHR$(B):NEXT J:ON S1
<>S2 GOTO 999:NEXT K
450 ? "(K)aseta czy (D)ysk";:INPUT C$:C=
(C$="K"):ON 1=C=(C$="D") GOTO 450:IF C T
HEN B$="C":P=1:? "Magnetofon";:GOTO 460
OR I=8 TO 13:READ A:AUT$(I,I)=CHR$(A):NE XT I:?
"Dysk";
456 DATA 255,255,13,128,183,129,224,2,225,2,13,128
460 ? " gotowy (Return)";:INPUT C$:TRAP 499:CLOSE
#1:OPEN #1,8,128*C,B$:FOR I=P TO 187:PUT
#1,ASC(AUT$(I,I);:NEXT I
470 FOR I=1 TO 227:PUT #1,ASC(POL$(I,I)):NEXT I:FOR
I=1 TO 26:PUT #1,ASC(A$(I,I)):NEXT I:IF C THEN 490
480 FOR I=1 TO 6:READ A:PUT #1,A:NEXT I
490 CLOSE #1:? "Zrobione!":END
499 ? "Blad ";PEEK(195);"Jeszcze raz:";:GOTO 460
```

```
*** Listing 3 ***

300 A$="68A208AC039EF007CA90009ED0FA60CA BD00E29D009
E8AD0F660":S2=0:L=300 310 M=0:FOR J=1 TO 51 STEP
2:B=0:FOR I=J TO J+1:N=ASC(A$(I,I)):S2=S2+I*N:N=N-
48-7*(N>57):B=B+N*(1+15**i=J)):NEX I 320
M=M+1:A$(M,M)=CHR$(B):NEXT J:ON S2<> 80466 GOTO 999
```

```
*** Listing 4 ***

299 DIM A$(100),C$(1):C$=CHR$(34)
350 POKE 82,0:POKE 752,1:? CHR$(125):POS
ITION 0,4:? "32750 DIM SPC$(26):SPC$=";
360 ? C$;:FOR I=1 TO 26:? CHR$(27);A$(I,
I);:NEXT I:? C$;":RET."
370 ? "POKE842, 12:CLR:L.:END":POSITION 0
,2:POKE 842,13:NEW
```

każdorazowym wywołaniu zamienia „kier” (CON--TROL + przecinek) na spację lub odwrotnie. Jeśli więc Twój program zawierał np. sekwencję GRAP HICS 1:P0KE 756,158:X=USR (1763), to powrót do trybu 0 winien wyglądać tak: GRAPHICS 0:POKE 756,156:X=USR (1763), o ile chcesz znowu dysponować „serduszkiem”.

Na stronie 6 pozostają w ten sposób raptem 3 wolne bajty — tyle, co nic; moje procedury zajmują bowiem przestrzeń od 1536 (hex 600) do 1788 (hex 6FC). Trudno — nic za darmo! Jeśli w programie używasz innych procedur maszynowych korzystających ze strony 6, naturalnie nastąpi kolizja. W takim wypadku wybierz raczej wersję z przechowywaniem procedur w BASIC-owych tablicach znakowych.

III. JESZCZE O KIERZE I SPACJI

Jeśli przeczytałeś uważnie część drugą, możesz zapytać: jak zamieniać serduszko na spację bez wersji „boot”? Można tak:

a) napisz NEW,

b) wprowadź programy 3 i 4,

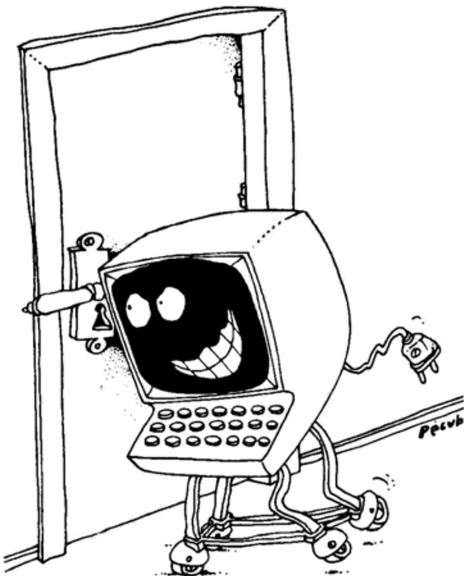
c) napisz RUN.

Otrzymasz króciutką procedurę w BASIC-u; możesz ją dołączyć do każdego swojego programu. Po odwołaniu się do niej (GOSUB 32750) możesz dowolną ilość razy rozkazywać X = USR (ADR (SPC$)). Efekt będzie analogiczny jak opisany wy--żej efekt komendy X = USR (1763) dla wersji „boot”. Naturalnie procedura w tablicy SPC$ ma sens tylko w połączeniu z użyciem procedury w tablicy POL$ (linie 32760-32764 stanowiące wynik działania programu z Bajtka 2/87).

IV. JAK ŁĄCZYĆ TE BLOKI?

Jeśli chcesz utworzyć „boot-cassette/disk” ze zredefiniowanymi klawiszami, wczytaj program z poprzedniego odcinka, wprowadź programy 1, 2 i 3, napisz RUN. Pomiń program 1, by otrzymać wersję „boot” bez redefinicji. Aby uzyskać zredefinio-wane klawisze w tablicy POLS, do starego programu dodaj tylko program 1. Pamiętaj: programy 1, 2 i 3 nie stanowią samodzielnej całości, lecz uzupełnienia programu z Bajtka 2/87. Tylko programy 3 i 4 stanowią razem program, który może (i musi) być wykonany osobno.

Krzysztof Leski



Co piszczy pod klawiaturą?

(cz. 3)

TABLICA ADRESÓW PROCEDUR SYSTEMOWYCH C.D.

Nr	Adres wektora	Adres rzeczywisty/opis 464	664	6128
07	BB15	1A7B	1C04	1C04
		Przydzielanie bufora dla stringów klawiszy funkcyjnych. Wej: — DE zawiera adres bufora, a HL jego długość. Wyj: — Jeżeli wszystko jest w porządku, wskaźnik CARRY jest „1”; jeśli nie CARRY jest „0”. A,BC,DE i HL są modyfikowane.		
08	BB18	1B56	1CDB	1CDB
		Oczekiwanie na znak pochodzący z klawiatury. Wej: — nie ma Wyj: — CARRY jest „1”, a akumulator zawiera znak wciśniętego klawisza. Wszystkie rejestry zachowane.		
09	BB1B	1C5C	1DE5	1DE5
		Testuje, czy występuje znak pochodzący z klawiatury. Wej: — nie ma Wyj: — Jeśli został wciśnięty klawisz, CARRY jest „1” i akumulator zawiera znak tego klawisza. Jeśli nie — Carry jest „0”.		
10	BB1E	1CBD	1E45	1E45
		Sprawdza, czy wciśnięty jest określony klawisz. Umożliwia również kontrolowanie drążków sterowych. Wej: — akumulator zawiera numer klawisza, który miał być sprawdzany. Wyj: — Jeżeli klawisz nie jest wciśnięty, ZERO jest „1”; jeżeli jest wciśnięty — ZERO jest „0” i akumulator oraz HL są modyfikowane, a C zawiera stan klawiszy SHIFT i CONTROL.		
11	BB21	1BB3	1D38	1D38
		Sprawdza, czy włączone są duże litery (klawisz CAPS LOCK). Wej: — nie ma. Wyj: — L zawiera stan SHIFT LOCK, a akumulator stan CAPS LOCK: 00 dla wyłączonego i FF dla włączonego.		
12	BB24	1C5C	1DE5	1DE5
		Odczyt stanu drążków sterowych. Wej: — nie ma. Wyj: — H zawiera stan drążka nr 0. L zawiera stan drążka nr 1. Akumulator zawiera stan drążka nr 0. Znaczenie poszczególnych bitów jest takie same, jak w funkcji „JOY” (patrz BASIC CPC).		
13	BB27	1D52	1ED8	1ED8
		Ustawia kod, który powinien być wygenerowany przy wciśnięciu pojedynczego klawisza (bez użycia CTRL lub SHIFT). Wej: — Akumulator zawiera numer klawisza. B zawiera kod ASCII tego klawisza. Wyj: — AF i HL są modyfikowane.		
14	BBA2	1D3E	1EC4	1EC4
		Dostarcza kod odpowiadający wciśniętemu klawiszowi. Wej: — Akumulator zawiera numer klawisza. Wyj: — Akumulator zawiera kod ASCII tego klawisza. HL i F są modyfikowane.		
15	BBD2	1D57	1EDD	1EDD
		Ustawia kod, który powinien być wygenerowany przy wciśnięciu jakiegoś klawisza jednocześnie z SHIFT. Wej: — Akumulator zawiera numer klawisza, a B kod ASCII tego klawisza. Wyj: — AF i HL są modyfikowane.		
16	BB30	1D43	1EC9	1EC9
		Dostarcza kod odpowiadający klawiszowi wciśniętemu razem z SHIFT. Wej: — Akumulator zawiera numer klawisza. Wyj: — Akumulator zawiera kod ASCII odpowiadający temu klawiszowi. HL są modyfikowane.		
17	BB33	1D5C	1EE2	1EE2
		Ustawia kod, który powinien być wygenerowany przy wciśnięciu jakiegoś klawisza jednocześnie z CTRL. Wej: — Akumulator zawiera numer klawisza. B zawiera jego kod ASCII. Wyj: — AF i HL są modyfikowane.		
18	BB36	1D48	1ECE	1ECE
		Dostarcza kod odpowiadający klawiszowi wciśniętemu razem z CTRL. Wej: — Akumulator zawiera numer klawisza. Wyj: — Akumulator zawiera kod ASCII odpowiadający temu klawiszowi. HL są modyfikowane.		
19	BB39	1CAB	1E34	1E34
		Ustawia powtarzanie klawisza. Wej: — Akumulator zawiera numer klawisza.Jeżeli klawisz może być powtarzany, B zawiera FF, jeśli nie — B zawiera 00. Wyj: — AF,BC i HL są modyfikowane.		
20	BB3C	1CA6	1E2F	1E2F
		Sprawdza, czy klawisz, którego numer jest dostarczany, może być powtarzany, czy nie. Wej: — Akumulator zawiera numer klawisza. Wyj: — Jeżeli klawisz może być powtarzany, wskaźnik O jest fałszem. Gdy klawisz nie może być powtarzany, wskaźnik O jest prawdą. W każdym razie CARRY jest O a AF i HL są modyfikowane.		

Wojciech Ziółek

```
PROGRAM 1

 1) 215e79227b5c3e07328d5ccd041e
 2) 6b0d3e00cd97223e02cd01160360
 3) 21ea78cd9f77e5cda77721490 6a0
 4) 790603cdbe77b828f232147d0519
 5) e1cd9f77cda77706053e24160532
 6) 08bb28059210fa18ef78321e045b
 7) 7dcd9f77cda7773e05bb28070578
 8) 3e08bb280218f132287d3a1e0363
 9) 7d2139ff11c800193d20fc3a045b
10) 287dfe08200119222 37daf06035c
11) 0721097d772310fc3ccd01160374
12) af32097d474fed430a7dcd6b04ec
13) 0dcdf27621c778cd9f77ed4b06bd
14) 0c7d21e603ed42dadf77cd3205f1
15) 77ed4b0e7d21e603ed42dadf062c
16) 77cd42772e55cd137822017d0478
17) 2eaacd137822047de60332030 3f1
18) 7d21007d0602c53e02b820140314
19) 3a147dfe23200821007dcd8003ff
20) 771817cdac7618123a147dfe0488
21) 1c2005cdcf76180621037dcd03df
22) 8077ed4b237d0b78b120fbcd05eb
23) 6777462b4ec5cd8e76c1c25b0611
24) 78cddf223e16d73e01d73e1d04e2
25) ed4b0a7dc503cd0c7782d7c105f1
26) 21097d7e34fe0a20113600210 2e9
27) e703ed4228080 3ed430a7dcd04d0
28) 1b1a3a287dfe052003cd597703d7
29) cd1b78c121037d05c20276c304c4
30) fd75cdaa2247047e0710fde605ce
31) 01c90100afaf77c93e0177c904e8
32) 3e0277c93e0377c93efbdbfe0613
33) e601cca8763efddbfee601cc0798
34) a4763efedbfee61ffe1dcc9d07b8
35) 76fe1bcca076c93edfdbfee60816
36) 01cca8763ebfdbfee601cca40718
37) 763e7fdbfee61ffe1bcc9d760709
38) fe1dcca076c9474fcddf2221064b
39) 8678afb6c85e2356234e234604dc
40) 23e5cdba24e118ee160221630536
41) 00ed4230031600c9210900ed0358
42) 42d01601c93e01b828102a0c0357
43) 7ded4b0a7d09220c7dc93e120409
44) 180e2a0e7ded4b0a7d09220e02d3
45) 7dc93e07f5c53e16d73e01d70586
46) c1cd0c77f1c582d7c1cd1b1a06e3
47) c9110a003a627de61f626fcd04a0
48) b503c906007e23b820013504033a
49) b82001340423b82001350 4b802fe
50) 200134c91600d5e5cdb777c505ae
51) e5cd6777e1cdb777cd8e76c107fe
52) e1e523712370e12029d1c97e062f
53) 23fe00c8d718f8e52b7cb5200631
54) fbcd8e023effbb28f8e1c9e507ff
55) 234e2346e1c97ebbc82310fa05b2
56) afc9ed5feeffcb1fcb1f9a570776
57) 3a017daa577e82e60377d11504ff
58) 20a4c9f5cd6b0d3e02cd011604eb
59) 214e79cd9f77f1fe00c2f87706eb
60) 3e3218023e31d7012c01cd3d0308
61) 1fc33675ed5f5fe60f571a1304b1
62) b838fbb930f8c9017d32cd040616
63) 7867c9011401cd4b78b8d8b90597
64) d00e1e57cd4b78b8d8b9d05f065b
65) ed5ffe05d0d53e02cd0116d105e9
66) 3e16d77ad77bd73e90d7afcd06ef
67) 0116c92a627d233e3fbc20020367
68) 260022627d7ec9066411010002ea
69) ed5f6fe60367c5d5e5cdb503070f
70) e1d1c110f5c1cd2177c3b475078a
71) f5c1cd319ec3c49ccd319ec307d4
72) c49c0101006e01010505ff0102dc
73) 050501010037010196 0001ff01db
74) 05050101050501015f0001ff0177
75) 003cffff050501ff050501ff044e
76) 0069ff017800ff010505ffff04e9
77) 0505ff017d0000475241435a02fe
78) 313a303030204752414 35a3202c4
79) 3a30303020535441574b413a02ef
80) 30303000000016010a574f4a01a1
81) 54454b204249416045 4b160302d9
82) 0842266220534f4654574152031 8
83) 45203139383 60d160908494c0206
84) 552047524143 5a59205b302d031d
85) 325d3f00160c09535a59424b028c
86) 4f5343205b312d355d3f001602a5
87) 0f0a445a5749454b205b542f02e5
88) 4e5d3f0000241c230000160b016e
89) 0a5759475241 4c2047524143031d
90) 5a00245a996666995a0000000330
```

Bzzz

Program „BZZZ” jest wersja znanej gry „Light CYCLE”. Dla tych, którzy jej nie znają — w skrócie zasady gry.

Przeznaczona jest ona w zasadzie dla dwóch osób, choć jedną z nich, a nawet obydwie, może zastąpić komputer. Każdy z graczy steruje punktem, przepraszam wspaniałym, transgalaktycznym światłolotem, który poruszając się zostawia za sobą ślad (oczywiście smugę światła z fotonowego silnika). Celem każdego z graczy jest jak najdłuższe unikanie zderzenia ze śladem własnym, śladem pozostawionym przez przeciwnika, brzegiem ekranu i pojawiającymi się od czasu do czasu „przeszkadzajkami” oraz utrudnianie tego samego przeciwnikowi poprzez zajeżdżanie mu drogi, spychanie na brzeg ekranu czy zamykanie w małych obszarach. Począwszy od startu rośnie stawka, która przypada zwycięzcy, tzn. graczowi, który zdołał dłużej uniknąć zderzenia.

Gra kończy się, gdy jeden z graczy uzyska 1000 punktów. Sterowanie odbywa się z klawiatury: dla gracza pierwszego są to klawisze „Z”, „X”, „Q”, „A”, a dla drugiego klawisze „M”, „Symbol Shift”, „P” i „ENTER”. Na początku każdej gry trzeba określić ilu graczy weźmie w niej udział, prędkość poruszania się punktów oraz włączyć lub wyłączyć dźwięk.

Gra „BZZZZ” (nazwa pochodzi od dźwięku wydawanego przez silniki kosmolotów) jest jeszcze mniej skomplikowana niż „Jumping Jack”, ale moim zdaniem bardziej wciągająca. (Pod warunkiem, że grają dwie osoby, ponieważ komputer nie jest zbyt ciekawym partnerem).

Jeżeli nie posiadasz „LIGHT CYCLE” a zainteresował Cię opis „BZZZZ”, pierwsze co musisz zrobić, to wprowadzić do komputera „Program 2” i nagrać go na taśmę instrukcjami:

LET a$ = CHR$ 22 + CHR$ 1 + CHR$0 + „BZZZZ” + CHR$ 6: SAVE a$ LINE 10

Wprowadź teraz do komputera (usuwając część właśnie nagraną na taśmę) „Program 3” i uruchom go instrukcją RUN. Będziesz teraz pytany 90 razy o linijki kodu maszynowego w zapisie heksadecymalnym (szesnastkowym), który umieszczony jest w „Programie 1”. Każda linijka składa się z 28 znaków, z których pierwsze 24 to 12 bajtów programu, a ostatnie 4 to suma kontrolna (suma 12 bajtów w jednej linijce), która umożliwia wykrycie błędów.

Po wprowadzeniu całości właściwy program nagrywany jest automatycznie za już umieszczoną na taśmie częścią „BZZZZ”.

Wojtek

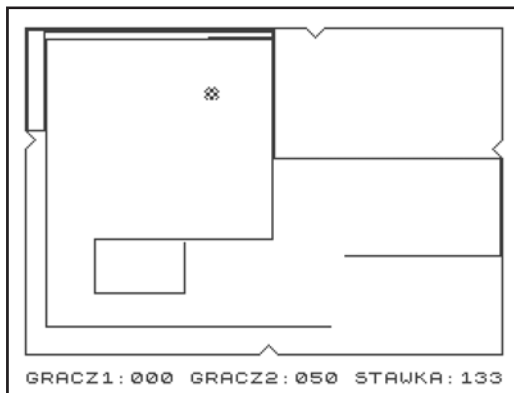
```
PROGRAM 2

10 BORDER 0: POKE 23693,0:
CLS : FOR N=0 TO 30 STEP 5:
BEEP .02,N: PRINT INK 7:
BRIGHT 1: FLASH 1: PRINT AT
18,8;"Loading BZZZ": LOAD
"" CODE : POKE 26393,7:
RANDOMIZE USR 30000
```

```
Program 3

3000 FOR n=0 TO 89
3010 PRINT AT 21,0;"Podaj
";n+1;" linijke": INPUT LINE
a$
3020 IF LEN a$<>28 THEN BEEP
1,-30: GO TO 3010
3030 LET s=0: FOR m=1 TO 24
STEP 2: LET b$=a$(m TO m+1)
3040 GO SUB 3100: POKE
29999+n*12+(m+1)/2,l
3050 LET s=s+l: NEXT m
3055 LET b$=a$(25 TO 28)
3060 GO SUB 3100: IF s<>l THEN
BEEP 1,-30: GO TO 3010
3070 BEEP 1,12: CLS : NEXT n
3080 PRINT "W porzadku": SAVE
"BZZZZ MK" CODE 30000,1078:
PRINT : PRINT "Przewin tasme w
celu weryfikacji": VERIFY ""
CODE : STOP
3100 LET l=0: FOR q=1 TO LEN
b$
3110 LET t= CODE b$(q): LET
r=87
3120 IF t<58 THEN LET r=48
3130 LET l=l+(t-r)*16↑(LEN b$-
q)
3140 NEXT q: RETURN
```

OBRAZEK Z GRY



ZOSTA

Od pewnego czasu interesowała mnie możliwość realizacji filmów animowanych na komputerze tak prostym, jak Spectrum. Niniejszy program jest efektem tych poszukiwań.

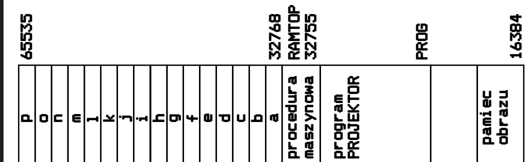
W poszukiwaniu rozwiązania zabrałem się do obliczeń. Ile obrazów możemy pomieścić w 48 KB RAM? To proste: pamięć obrazu zajmuje 6912 bajtów; BASIC daje użytkownikowi nieco ponad 40 KB. Daje to w sumie 6 „ekranów” plus jeden w swoim zwykłym miejscu — pod adresem 16384. Razem siedem kadrów filmu, z których każdy zajmuje cały ekran.

Gdybyśmy jednak wykorzystywali jako klatkę filmu tylko część ekranu ...Pytanie, jak ową część ekranu wybrać staje się proste po przyjrzeniu się strukturze odwzorowania obrazu w pamięci. Mianowicie z 6912 bajtów równo 6 KB zajmuje rysunek, reszta — to kolor atrybuty obrazu. Z kolei pamięć rysunku podzielona jest na 3 sektory po 2 KB. Widać je wyraźnie podczas wczytywania obrazków z magnetofonu. Zrezygnowałem z koloru i odrzuciłem dwa sektory ekranu, zamierzając w charakterze klatki filmu wykorzystać tylko środkowy. W ten sposób da się pomieścić 16 kadrów w 32 KB, przy czym pozostaje jeszcze dość miejsca na ewentualną rozbudowę programu oraz swoboda w wykorzystywaniu ekranu poza „projekcją”.

16 kadrów umieściłem w górnych 32 KB pamięci, nazywając je kolejnymi literami alfabetu (a..p).

Zadanie przepisania zawartości kadru na ekran pełni krótka procedura w języku maszynowym. Zainteresowanych jej działaniem odsyłam do artykułu „jak schować obrazek” z „Bajtka” nr 8/86. Kodu maszynowego użyłem oczywiście przez konieczność maksymalnego przyspieszenia działania procedury.

Rys.1 przedstawia mapę pamięci dla potrzeb naszego programu (zauważmy, że straciliśmy obszar grafiki użytkownika — UDG).



Rys.l. Organizacja pamięci w programie „Projektor”.

Funkcje programu wywoływane są przy pomocy menu ukazującego się w dwóch dolnych liniach ekranu.

Po uruchomieniu programu ukazuje się nam

MENU PODSTAWOWE

PARAMETRY — otwarcie menu parametrów projekcji

KLAPS! — uruchomienie projekcji

EKRAN — usunięcie treści menu z ekranu. Wszystkie funkcje menu są nadal dostępne poprzez naciśnięcie odpowiednich klawiszy

TAŚMA — otwarcie menu współpracy z magnetofonem

ZAPAMIĘTANIE — zapisanie aktualnej zawartości środkowego sektora ekranu jako jednej z klatek (a..p). Program pyta o oznaczenie klatki

BREAK — przerwanie projekcji filmu.

MENU „PARAMETRY”

SEKWENCJA — zaprogramowanie kolejności kadrów, w której będą one wyświetlane podczas projekcji Standardowo SEKWENCJA = „abcdefghijklmnop”

Ń FILMOWCEM

ZAPĘTLENIE — zapętlenie taśmy; projekcja sekwencji kadrów jest wyświetlana „w kółko".
ROZCIĘCIE — odwrotnie do funkcji ZAPĘTLENIE; sekwencja kadrów jest wyświetlana jednorazowo
TEMPO — zmiana szybkości projekcji (zwiększenie wartości parametru TEMPO powoduje jej zwolnienie). Podanie TEMPO=0 pozwala na oglądanie poklatkowe filmu. Standardowo. TEMPO = 1
MENU — powrót do menu podstawowego.

MENU „TAŚMA"

ŁADOWANIE KADRU — wczytanie z taśmy zawartości obrazu.
ŁADOWANIE FILMU — wczytanie z taśmy zawartości wszystkich klatek filmu, nagranych uprzednio przy pomocy funkcji NAGRANIE FILMU.
NAGRANIE FILMU — nagranie na taśmę obszaru pamięci, zawierającego wszystkie 16 klatek filmu.
MENU — powrót do głównego menu.

Tak zbudowany projektor jest jedynie narzędziem pozwalającym na wyświetlenie filmu; samo jego stworzenie jest zajęciem dość trudnym i żmudnym. Praca nad nim wyglądać powinna w następujący sposób:

a. przygotowujemy kolejno 16 (lub tyle, ile nam potrzeba) rysunków, nagrywając je kolejno na taśmie. Możemy się w tym celu posłużyć szeroką gamą oprogramowania. Ja stosowałem THE ARTIST i LOGO.

b. po wczytaniu 1 uruchomieniu projektora nagrywamy z kolei przy jego pomocy przygotowane obrazki z taśmy i zapamiętujemy je jako kolejne klatki.

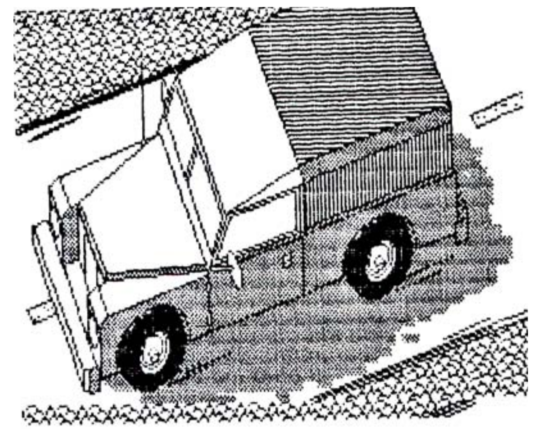
c. Programujemy sekwencję, tempo i zapętlenie taśmy — i oglądamy efekt na ekranie.

d. Jeżeli chcemy film zachować sobie na przyszłość, nagrywamy go na taśmę przy pomocy odpowiedniej funkcji programu.

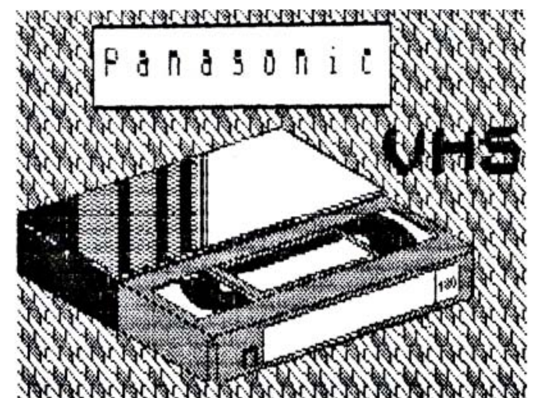
Jakkolwiek stworzenie w wyżej opisany sposób dobrze animowanych, efektownych obrazów wymaga sporej cierpliwości, wyniki potrafią nieraz wynagrodzić trud. Polecam też zapoznanie się z zamieszczonym w tym numerze artykułem poświęconym LOGO — inspiracją i pierwszym zastosowaniem zamieszczonych tam rozwiązań były właśnie filmy animowane.

Marcin Waligórski

Oto dwa różne kadry filmu, w którym widoczny napis jest animowany w przestrzeni. Wszystkie stadia ruchu zostały zaprogramowane i wykreślone w LOGO, a następnie zacieniowane przy pomocy programu THE ARTIST.



Rys.3. Oto przykład zbudowania filmu na całym ekranie. Trick polega na tym, że tylko elementy obrazu leżące w środkowym sektorze (pasy na jezdni, cień samochodu), są ruchome. Film powstał przy pomocy THE ARTIST.



Rys.4. Tutaj nie dało się postąpić tak, jak na rys.S. Kaseta VHS, poruszana w filmie, zajmuje dwa sektory ekranu, za to film wymagał niewielu klatek. Prosta modyfikacja programu spełniła te wymagania. Zarówno publikowana ilustracja, jak też wszystkie inne fazy ruchu widocznego przedmiotu powstały przy pomocy THE ARTIST z zastosowaniem techniki OVER-LAY.



Rys.5. To wygląda jak okładka gry COMMANDO, lecz nią już nie jest – można porównać! Niemal wszystkie postacie widoczne na rysunku – łącznie z żołnierzem na pierwszym planie – zostały wprawione w ruch. Dodatkowe efekty strzałów i wybuchów powodują, że na ekranie toczy się bitwa „jak prawdziwa". Pomocny okazał się znów THE ARTIST (np. wybuchy zostały wykreślone techniką PLOT-POINT).

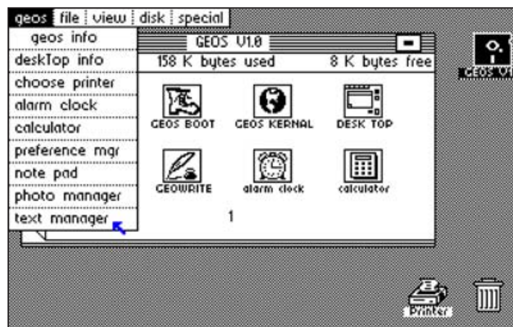
```
1 REM +------------------------+
2 REM +    Projektor filmow    +
3 REM +      animowanych       +
4 REM +    MARCIN WALIGORSKI   +
5 REM +------------------------+
7 REM +------------------------+
8 REM +     Inicjalizacja      +
9 REM +------------------------+
10 CLEAR 32755
12 LET Petla=0: LET Ekr=1
13 LET Tempo=1
15 LET S$="abcdefghijklmnop"
16 BORDER 7: PAPER 7: INK 0: CLS
17 REM +------------------------+
18 REM +  Procedura maszynowa   +
19 REM +------------------------+
20 FOR k=32756 TO 32767
30 READ bajt: POKE k,bajt
40 NEXT k
50 DATA 33,0,128,17,0,72
60 DATA 1,0,8,237,176,201
107 REM +------------------------+
108 REM +      Obsluga menu      +
109 REM +------------------------+
110 PRINT AT 10,2;"PROJEKTOR
FILMOW ANIMOWANYCH"
120 PRINT #0;AT 0,0;"
"
125 IF Ekr THEN PRINT #0,AT
0,0;"Parametry  Klaps!
Ekran Tasma      Zapamietanie
Break "
130 LET A$=INKEY$
135 IF A$<>"" THEN BEEP .1,30
140 IF A$="t" THEN GO TO 200
156 IF A$="e" THEN GO TO 300
160 IF A$="p" THEN GO TO 400
166 IF A$="k" THEN GO TO 600
176 IF A$="z" THEN GO TO 500
190 GO TO 130
197 REM +------------------------+
198 REM +  Obsluga magnetofonu   +
199 REM +------------------------+
200 PRINT #0,AT 0,0;"Ladowanie
Kadru/Filmu            Nagranie
Filmu         Menu        "
205 LET A$=INKEY$
207 IF A$<>"" THEN BEEP .1,30
210 IF A$="m" THEN GO TO 120
220 IF A$="k" THEN GO SUB 270:
LOAD N$ SCREEN$ : GO TO 120
230 IF A$="f" THEN GO SUB 270:
LOAD N$ SCREEN$ : GO TO 120
240 IF A$="n" THEN GO SUB 270:
SAVE N$ CODE 32768,32768: GO TO
120
250 GO TO 205
267 REM +------------------------+
268 REM +  pytanie o nazwe pliku +
269 REM +------------------------+
270 INPUT "Nazwa pliku ?",N$
280 RETURN
297 REM +------------------------+
298 REM +      Tryb ekranu       +
299 REM +------------------------+
300 LET Ekr=NOT Ekr
310 IF NOT Ekr THEN PRINT #0,AT 0,
0;"
320 PAUSE 30
330 GO TO 120
397 REM +------------------------+
398 REM +   Zmiana parametrow    +
399 REM +------------------------+
400 PRINT #0,AT 0,0;"Sekwencja
Tempo                Zapentlenie
Rozciecie     Menu  "
405 LET A$=INKEY$
410 IF A$<>"" THEN BEEP .1,30
420 IF A$="m" THEN GO TO 120
430 IF A$="s" THEN INPUT
"Sekwencja kadrow ?";S$: GO TO 400
440 IF A$="t" THEN INPUT "Tempo
(0... ) ?";Tempo: GO TO 400
450 IF A$="z" THEN LET Petla=1: GO
TO 400
460 IF A$="r" THEN LET Petla=0: GO
TO 400
470 GO TO 405
497 REM +------------------------+
498 REM +   Zapamietanie kadru   +
499 REM +------------------------+
500 INPUT "Oznaczenia kadru
(a...p)? ";Z$
510 IF Z$<"a" OR Z$>"z" THEN GO TO
500
515 LET Nr= CODE (Z$)-97
530 POKE 32761,128+8*Nr
540 RANDOMIZE USR 32756
550 GO TO 120
597 REM +------------------------+
598 REM +    Projekcja filmu     +
599 REM +------------------------+
600 POKE 32761,72
610 LET A$=S$
620 :
630 LET Nr= CODE (A$)-97
646 POKE 32758,128+8*Nr
650 RANDOMIZE USR 32756
655 BEEP .0002,45: PAUSE Tempo
660 LET A$=A$(2 TO LEN (A$))
665 IF INKEY$="b" THEN GO TO 120
670 IF A$<>"" THEN GO TO 630
680 :
690 IF Petla THEN GO TO 610
700 GO TO 130
```

GEOS

czyli jak C-64 może dogonić McIntosha

GEOS, czyli Graphic Environment Operating System (system operacyjny porozumiewający się z użytkownikiem na drodze graficznej), jest systemem nowej generacji, przeznaczonym dla tych, którzy pragną możliwie szeroko wykorzystywać komputer, a równocześnie nie mają ochoty na żmudne studiowanie instrukcji obsługi programów użytkowych.

C-64 pracujący pod kontrolą systemu GEOS do złudzenia przypomina komputery stojące o klasę wyżej, jak np. ATARI 520 ST, McIntosh czy AMIGA. Charakterystyczne obrazkowe menu pozwala na wybranie dowolne] funkcji bez dotykania klawiatury. Posługujemy się drążkiem sterowym (joystickiem) lub „myszą". Na firmowym dysku, prócz systemu operacyjnego znajduje się szereg programów użytkowych GeoWrite, GeoPaint, Alarm Clock, Note Pad i Calculator. Każdy z tych programów jest doskonale opracowany i zasługuje na osobne omówienie.



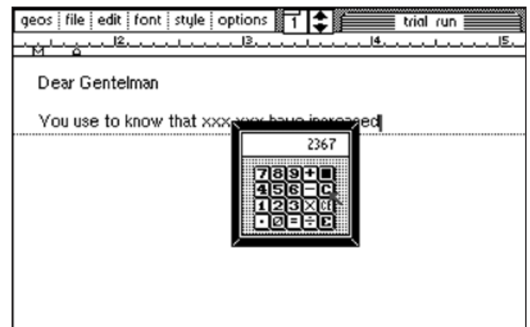
Rys.1 GEOS przeznaczony jest dla ludzi, którzy nic mają ochoty na żmudne studiowanie instrukcji.

Alarm Clock, czyli po prostu budzik, pozwala na uniknięcie wielu nieprzyjemności związanych z „zasiedzeniem" nad komputerem. Dysponujemy zegarem czasu rzeczywistego z możliwością alarmu o ustalonej godzinie.

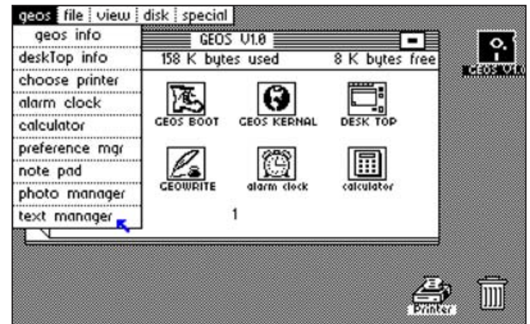
Calculator — tej nazwy nie trzeba tłumaczyć. Po wywołaniu programu na ekranie ukazuje się cztero-działaniowy kalkulator. Cyfry wprowadzamy z klawiatury bądź „myszą". Zaletą tego programu jest fakt, iż można wywołać go w dowolnej chwili, nawet podczas korzystania z dowolnego innego programu użytkowego.

Note Pad to po prostu notes, wykorzystujący do przechowywania danych dysk systemowy. Pojemność notesu wynosi 127 stron po 255 znaków.

Następne dwa programy wymagają nieco więcej uwagi. GeoPaint jest doskonałym programem graficznym, pozwalającym tworzyć obrazy w 16 kolorach, przy wymiarach 640 x 800 punktów.



Rys. 2 Kalklator możemy przywołać na ekran w każdej chwili, niezależnie od tego, z którym programem systemu pracujemy



Rys. 3 Notes Note Pad może służyć do pozostawienia wiadomości domownikom (o ile przyzwyczaimy ich, aby po powrocie do domu włączali komputer).

Rozdzielczość taka możliwa jest do zrealizowania tylko w przypadku zastosowania odpowiedniej drukarki (GEOS potrafi obsłużyć praktycznie każdy model). Jeśli pracujemy na ekranie, możemy w obrębie całego rysunku dowolnie umieścić okno 320 x 200 punktów i w nim pracować. GeoPaint oferuje wiele przydatnych narzędzi graficznych, różne wzory do wypełniania zamkniętych przestrzeni itp. Możliwe jest też korzystanie z banku gotowych rysunków — niestety, bank taki nie jest dostarczany przez producenta, trzeba go samemu stworzyć. Pod względem łatwości uży-

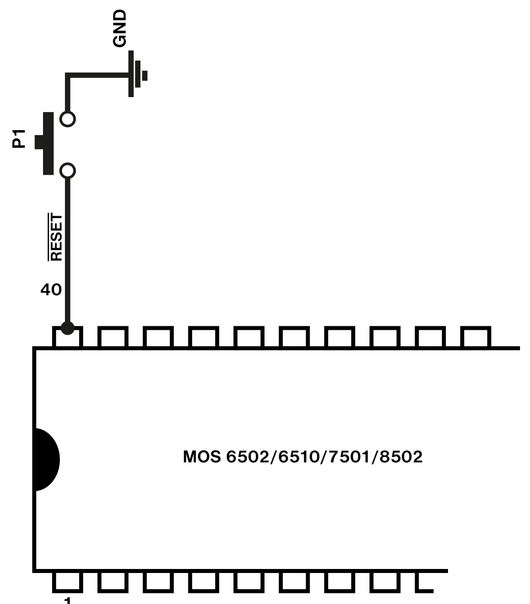
PRZYCISK RESET DLA C-64

Czytelnikom mało doświadczonym w sprawach elektronicznych odradzamy samodzielny montaż tego przycisku i proponujemy dokonać ww. przeróbki pod okiem doświadczonego fachowca.

Jednym z wielu udogodnień zastosowanych w komputerach PLUS/4, C-16, 116 i 128 jest przycisk RESET, który umożliwia wyzerowanie komputera bez konieczności wyłączania go z sieci. Zazdrośników — posiadaczy Commodore 64 pocieszamy, że z łatwością mogą sami wyposażyć swój komputer w ten wygodny guziczek. Warto także wspomnieć, że używanie tego przycisku jest znacznie bardziej bezpieczne aniżeli wyłączanie komputera z sieci — sprzęt nie jest wtedy narażony na ewentualne przepięcia, a także oszczędza się wyłącznik sieciowy.

Zasada działania przycisku RESET polega na chwilowym zwieraniu z masą wejścia **reset** mikroprocesora (patrz rys.1) co spowoduje jego całkowite wyzerowanie. Ponieważ wyprowadzenie procesora jest połączone z innymi układami (np. 6526 — Complex Interfa-ce Adapter) — wciśnięcie przycisku RESET powoduje wyzerowanie całego komputera. Odpowiednikiem programowym opisanej tu przeróbki jest SYS 64738

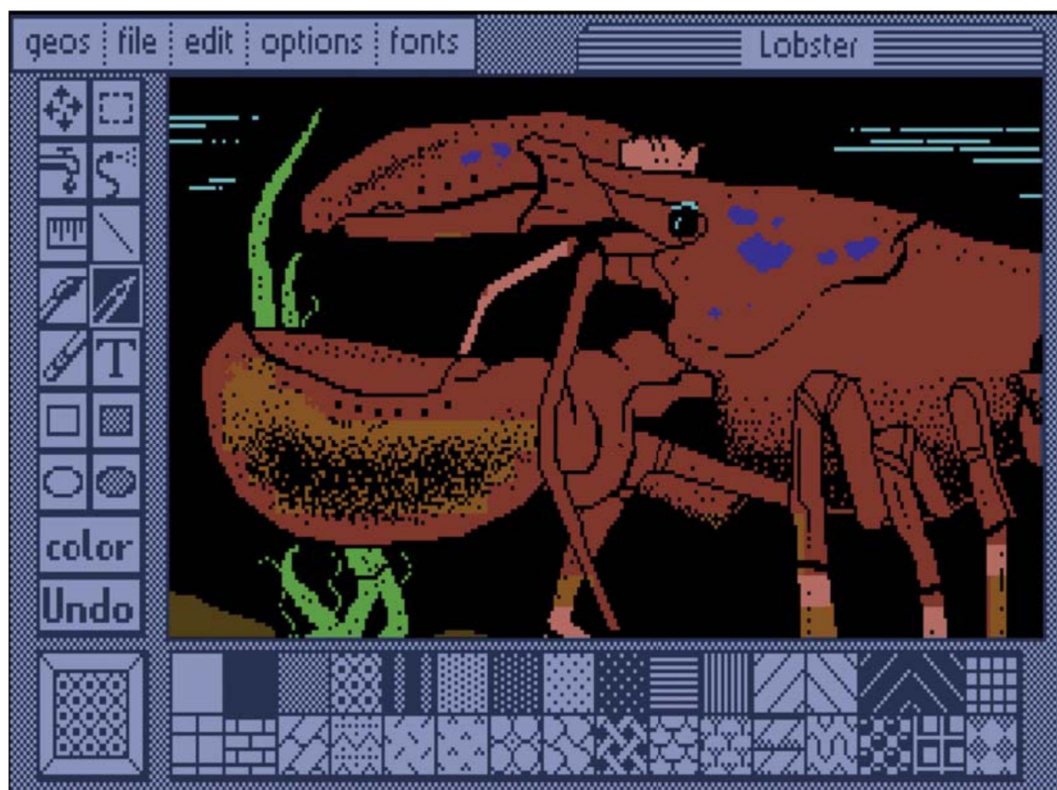
Do wykonania urządzenia potrzebne nam będą następujące materiały i narzędzia: dowolny przełącznik astabilny (może być nawet wykonany z małego isostatu), dwa długie izo-



RYSUNEK 1 – Zasada działania przycisku RESET

lowane przewody (najlepiej linka), lutownica o mocy do 25 W, cyna i kalafonia.

Skoro znamy już zasadę działania możemy się zastanowić w jaki sposób zamontować nasz przycisk, gdyż możliwe są tu dwa rozwiązania: albo montaż na stałe w obudowie, albo dodatkowa przystawka. W tym drugim przypadku należy się zastanowić, do które-

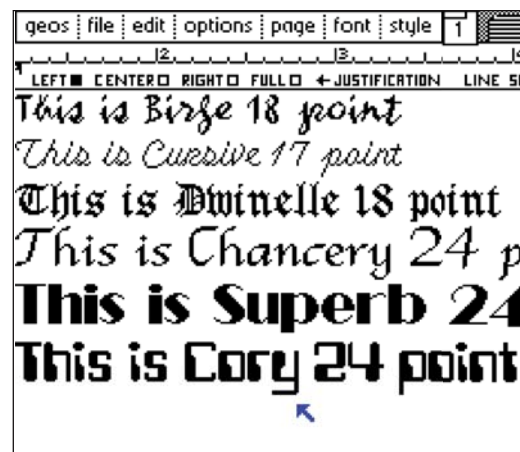


Rys.4. GeoPaint pozwala tworzyć obrazy w 16 kolorach (640 x 800 punktów).

cia i możliwości GeoPaint bije na głowę starsze programy typu Koalapainter czy Mi-croillustrator. Całość podobna jest do apple'ows-kiego Macpaint, tyle, że przy dwukrotnie mniejszej rozdzielczości oferuje 16 kolorów, których MacPa-intowi brakuje.

Przebojem GEOS-a jest GeoWrite, procesor tekstu pracujący w trybie graficznym. Pomijając typowe dla takich programów cechy jak funkcje wstawiania, usuwania czy wyszukiwania słów, warto opisać szatę graficzną powstającego tekstu. Mamy do dyspozycji kilkadziesiąt krojów pisma w różnych rozmiarach, możliwość korzystania z banku rysunków, stworzonego przy pomocy GeoPaint, a w efekcie — możliwość przygotowania tekstu, który po wydrukowaniu nie ustępuje pracom przygotowanym przy pomocy specjalizowanych programów typu NEWSROOM.

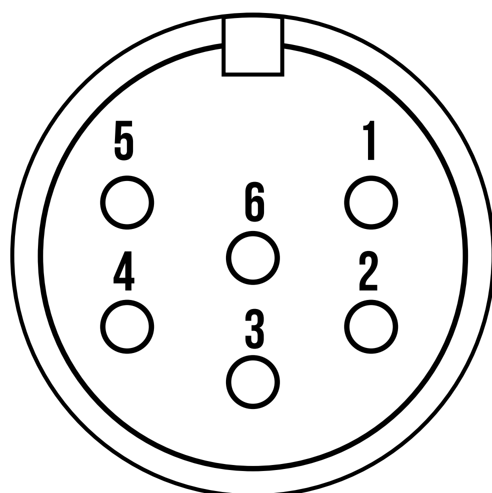
Poza istnieniem kilku wartościowych programów użytkowych. GEOS — jak przystało na do-



Rys.5 GeoWrite potrafi o wiele więcej niż przyzwyczailiśmy się wymagać od tradycyjnych edytorów tekstu.

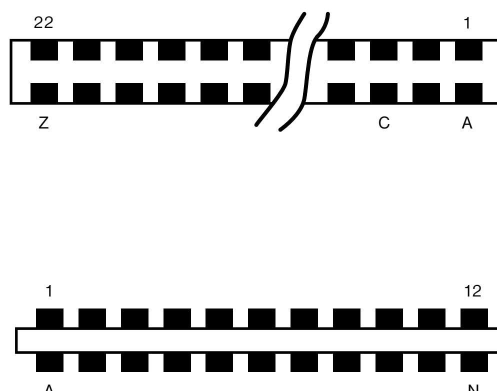
bry system operacyjny — przyspiesza znacznie (prawie dziesięciokrotnie) pracę ze stacją dysków, umożliwiając jednocześnie korzystanie z innych programów. Umożliwia pracę z dwiema stacjami dysków, daje też możliwość dostosowania stylu pracy do swoich upodobań — np. katalog dysku może być wczytany jako zbiór obrazków, jako standardowy katalog uporządkowany według nazw, długości lub rodzaju zbiorów, wreszcie w dowolny inny, zdefiniowany przez użytkownika sposób.

Systemy dążące do maksymalnego uproszczenia sposobu pracy z komputerem są przyszłością mikroinformatyki. Technika wykorzystująca okna, ikony i „myszkę" pozwala na używanie komputera bez instrukcji obsługi, przygotowania fachowego i jakiegokolwiek wykształcenia w tej dziedzinie. Miejmy nadzieję, że i w polskich komputerach kiedyś zobaczymy na ekranie okno z napisem „wybierz rysunek i naciśnij klawisz „myszki".



RYSUNEK 2 – Schemat wyprowadzeń portu szeregowego (SERIAL BUS) – widok z przodu

2 – GND *6 – RESET*



A, N, 1, 12– GND
3 – RESET

RYSUNEK 3 – Schemat wyprowadzeń portów: rozszerzającego (a) i użytkownika (b) – widok z przodu

go z portów będzie nasz przycisk dołączany — najlepszym wydaje się być port szeregowy (serial bus) z tym, że niektóre wersje C-64 nie są jak na złość łączone z linią **reset** w tym porcie. Jeżeli jednak połączenie takie istnieje, to wystarczy na odpowiednim sześcio-bolcowym wtyku osadzić mikrowyłącznik tak, aby zwierał po wciśnięciu bolce 2 i 6 i przycisk jest gotowy do użytku. W razie konieczności dołączenia stacji dysków czy drukarki, tak wykonany przełącznik umieszczamy po prostu w drugim wolnym gniazdku drukarki czy stacji.

Na rysunku 3 przedstawiono wyprowadzenia portów: rozszerzającego (expansion port) oraz użytkownika (user port). Montaż na stałe będzie łatwiejszy o tyle, że możemy przylutować przewody w dowolne miejsce ścieżki wyprowadzonej od końcówki 40 mikroprocesora (czy to przy porcie szeregowym, czy w jakimkolwiek innym miejscu). Polecamy odnalezienie najszerszego fragmentu tej ścieżki i lutowanie przewodu właśnie tam. Sam przycisk proponujemy wyprowadzić na tylną ściankę komputera, gdyż zabezpiecza to przed jego przypadkowym wciśnięciem.

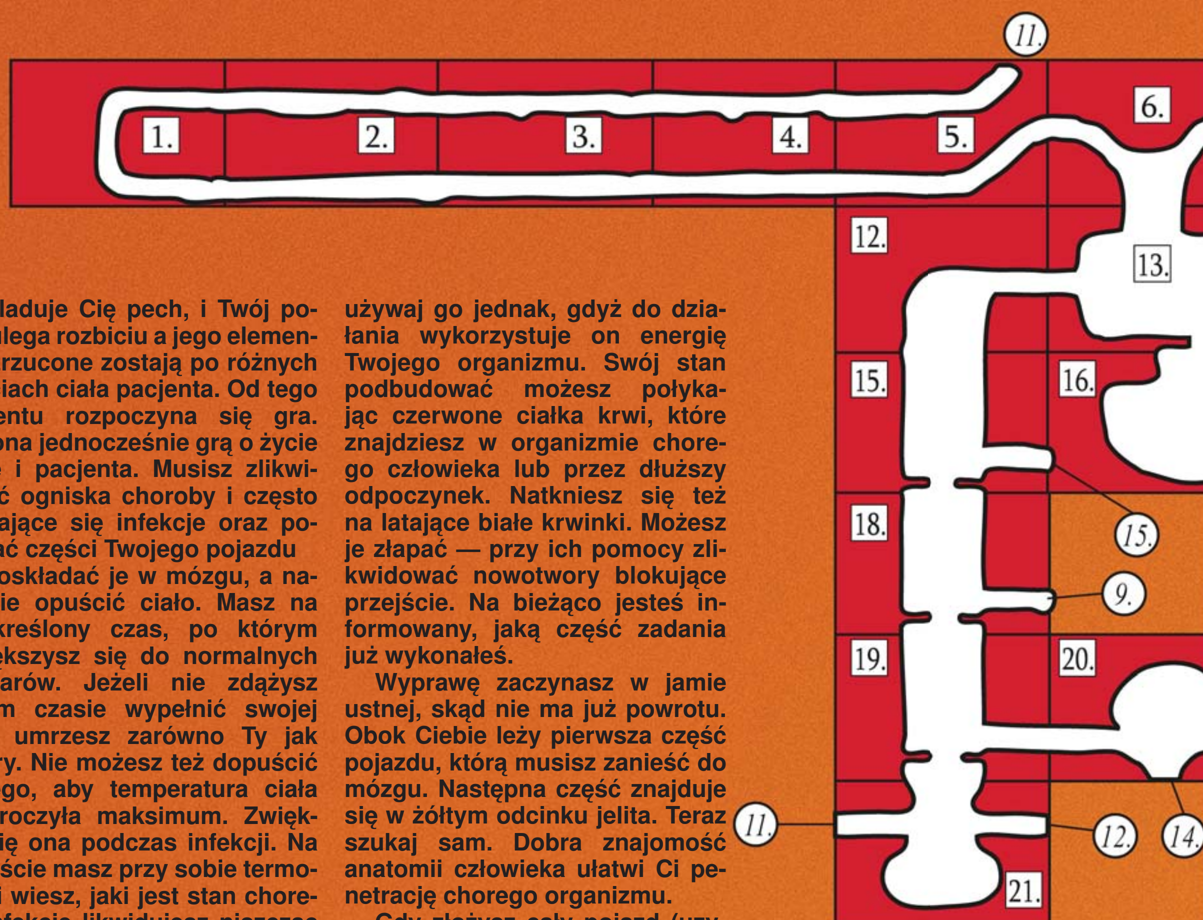
Opisane rozwiązanie pozwala na wyzerowanie komputera w 99% wypadków; tym niemniej mogą się zdarzyć sytuacje, że nawet wciśnięcie przycisku RESET nie spowoduje wyzerowania, co nie wynika z działania samego urządzenia lecz konstrukcji programu. Poprawność działania przycisku można sprawdzić w następujący sposób:

1. Uruchomić komputer;
2. Wykonać PRINT CHR$ (147) i RETURN;
3. wcisnąć przycisk RESET.

Obraz powinien zwęzić się na ok. 1 sekundę i powrócić do zgłoszenia się systemu, jak po uruchomieniu komputera.

Michał Silski
Klaudiusz Dybowski

FANTASTIC VOYAGE



Przyszedł czas., gdy rzeczywistość przerosła fantazję. Najnowszym wynalazkiem jest aparatura do zmniejszenia wszelkich obiektów. Ordynator największego na świecie szpitala postanowił wykorzystać tę aparaturę w medycynie. Przywieziono właśnie beznadziejnie chorego pacjenta. Choroba ogarnęła dużą część organizmu i dalej się rozprzestrzenia. Jedynym wyjściem jest zminiaturyzowanie jednego z lekarzy i umieszczenie go wewnątrz chorego. Wezwano więc Ciebie na pomoc. Nie zważając na niebezpieczeństwa postanawiasz pomóc choremu.

Wsiadasz w swój pojazd przystosowany do poruszania się po organizmie człowieka, zmniejszasz się odpowiednio wraz z nim i rozpoczynasz podróż w głąb chorego organizmu. Niestety prześladuje Cię pech, i Twój pojazd ulega rozbiciu a jego elementy rozrzucone zostają po różnych częściach ciała pacjenta. Od tego momentu rozpoczyna się gra. Jest ona jednocześnie grą o życie Twoje i pacjenta. Musisz zlikwidować ogniska choroby i często zdarzające się infekcje oraz pozbierać części Twojego pojazdu i poskładać je w mózgu, a następnie opuścić ciało. Masz na to określony czas, po którym powiększysz się do normalnych rozmiarów. Jeżeli nie zdążysz w tym czasie wypełnić swojej misji, umrzesz zarówno Ty jak i chory. Nie możesz też dopuścić co tego, aby temperatura ciała przekroczyła maksimum. Zwiększa się ona podczas infekcji. Na szczęście masz przy sobie termometr i wiesz, jaki jest stan chorego. Infekcję likwidujesz niszcząc drobnoustroje chorobotwórcze. Zadanie masz ułatwione, gdyż jesteś cały czas informowany, która część ciała jest zainfekowana.

Nawet wtedy, gdy chwilowo organizm nie wykazuje oznak infekcji, przeszkadzają Ci bakterie żyjące w symbiozie z organizmem ale groźne dla Ciebie. Poruszanie utrudniają też zatory w naczyniach krwionośnych a ruchome zastawki są w stanie Cię zniszczyć. Niektóre części ciała, np. ściany płuc też są groźne dla Ciebie.

Do obrony przed bakteriami i infekcjami a także do wystrzeliwania przejść w zatorach służy ręczny miotacz energii. Nie nadużywaj go jednak, gdyż do działania wykorzystuje on energię Twojego organizmu. Swój stan podbudować możesz połykając czerwone ciałka krwi, które znajdziesz w organizmie chorego człowieka lub przez dłuższy odpoczynek. Natkniesz się też na latające białe krwinki. Możesz je złapać — przy ich pomocy zlikwidować nowotwory blokujące przejście. Na bieżąco jesteś informowany, jaką część zadania już wykonałeś.

Wyprawę zaczynasz w jamie ustnej, skąd nie ma już powrotu. Obok Ciebie leży pierwsza część pojazdu, którą musisz zanieść do mózgu. Następna część znajduje się w żółtym odcinku jelita. Teraz szukaj sam. Dobra znajomość anatomii człowieka ułatwi Ci penetrację chorego organizmu.

Gdy złożysz cały pojazd (uzyskasz wynik 90 proc), musisz przez kanał nerwu ocznego i oko opuścić organizm i dostać się do lądowiska. Tam uzyskasz brakujące 10 proc. i otrzymasz gratulacje. Powodzenia!

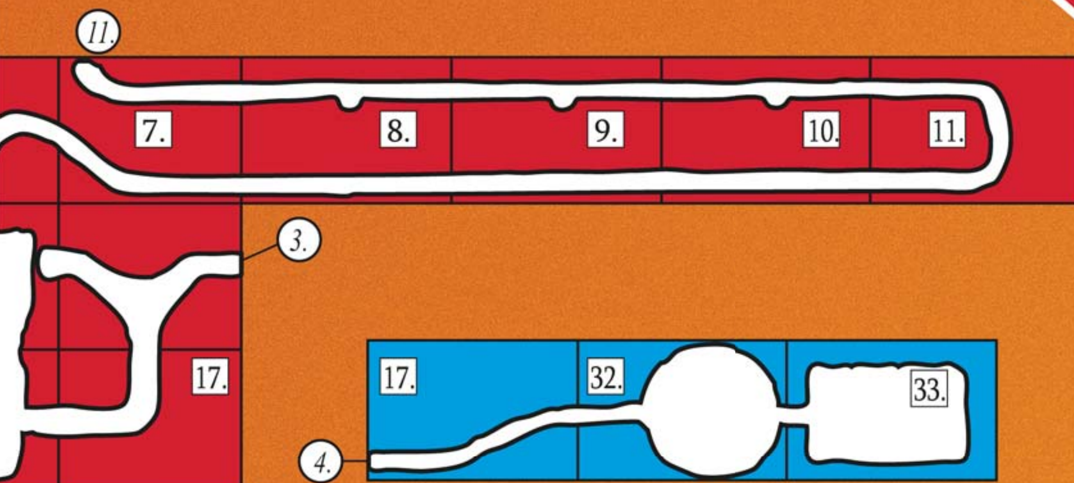
Marcin i Mateusz Przasnyski

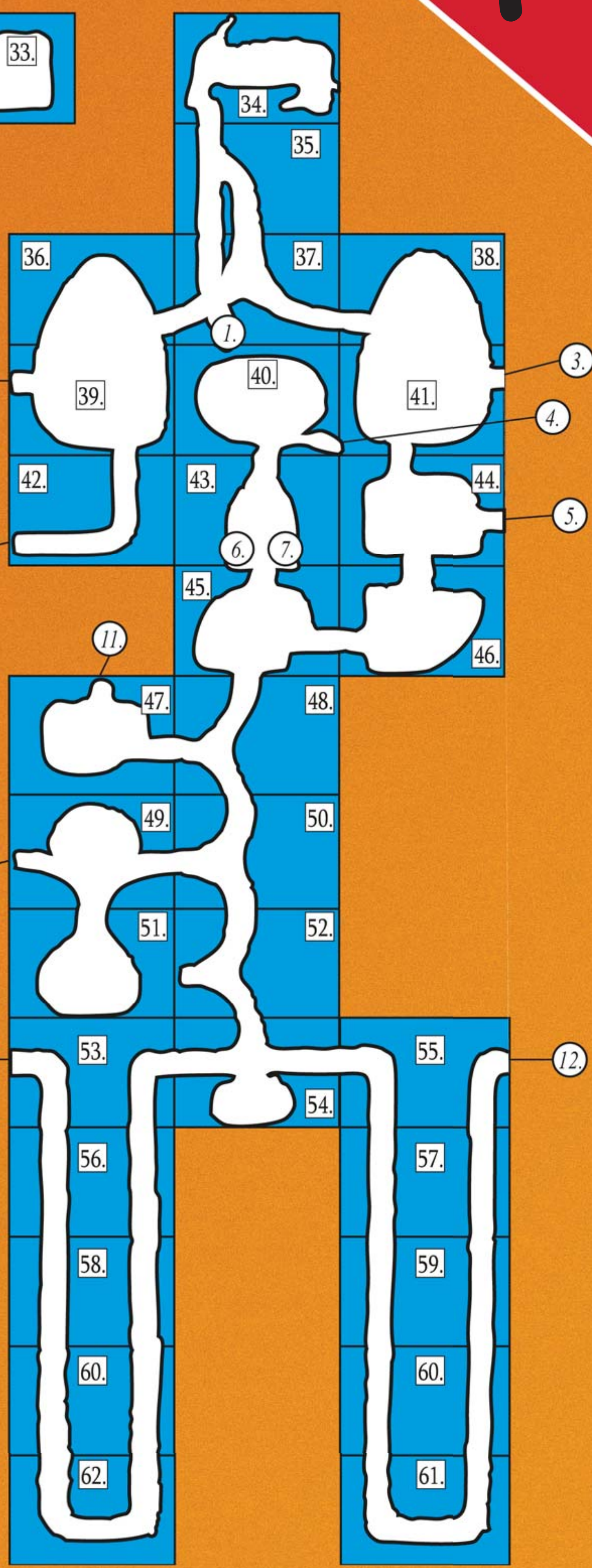
1. Right hand — prawa dłoń
2. Right forearm — prawe przedramię
3. Right elbow — prawy łokieć
4. Right arm — prawe ramię
5. Right shoulder — prawy bark
6. Superior vena cava — górna żyła główna
7. Left shoulder — lewy bark
8. Left arm — lewe ramię
9. Left elbow — lewy łokieć
10. Left forearm — lewe przedramię
11. Left hand — lewa dłoń
12. Interior vena cava — dolna żyła główna
13. Right atrium — prawy przed-J sionek
14. Pulmonary artery — tętnica płucna
15. Hepatic vein — żyła wątrobowa
16. Right ventricle — prawa komora
17. Pulmonary artery — tętnica płucna
18. Renal vein — żyła nerkowa
19. Renal vein — żyła nerkowa
20. Right kidney — prawa nerka



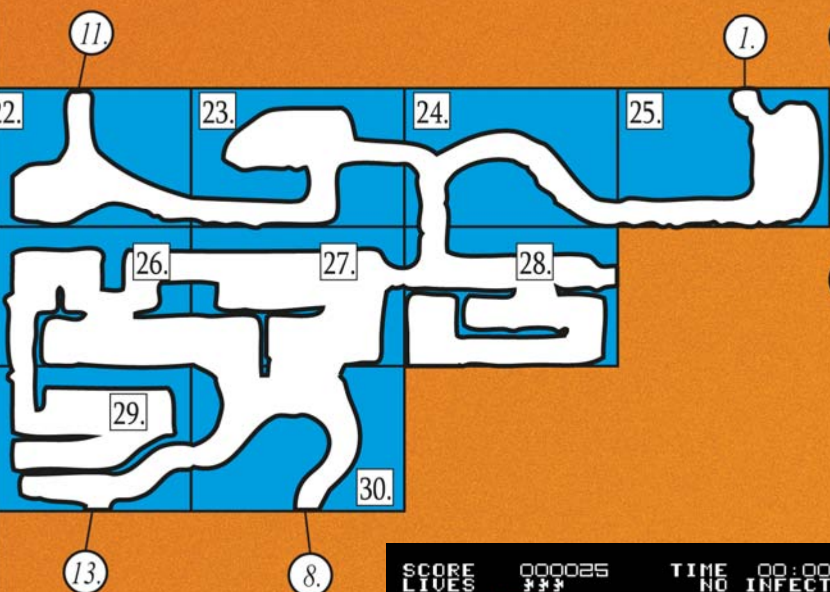
Fantastic voyage

FANTASTYCZNA PODRÓŻ





21. Interior vena cava — dolna żyła główna
22. Gall bladder — woreczek żółciowy
23. Pancreas — trzustka
24. Duodenal — dwunastnica
25. Stomach — żołądek
26. Liver — wątroba
27. Liver — wątroba
28. Intestine — jelito
29. Liver — wątroba
30. Liver — wątroba
31. Optic nerve — nerw oczny
32. Eye —oko
33. Landing bay — lądowisko
34. Mouth — jama ustna
35. Throat — gardło
36. Right lung — prawe płuco
37. Bronchi (Esophagus — oskrzela) przełyk
38. Lefl lung — lewe płuco
39. RIght lung — prawe płuco
40. Brain — mózg
41. Left lung — lewe płuco
42. Pulmonary vein — żyła płucna
43. Aorta — aorta
44. Left atrium — lewy przedsionek
45. Aorta — aorta
46. Left ventricle — lewa komora
47. Hepatic artery — tętnica wątrobowa
48. Hepatic artery — tętnica wątrobowa
49. Left kidney — lewa nerka
50. Renal artery — tętnica nerkowa
51. Bladder — pęcherz
52. Renal artery — tętnica nerkowa
53. Right pelvis — prawa część miednicy
54. Abdominal aorta — aorta brzuszna
55. Left pelvis — lewa część miednicy
56. Right thigh — prawe udo
57. Left thigh — lewe udo
58. Right knee — prawe kolano
59. Left knee — lewe kolano
60. Right calf — prawa łydka
61. Left calf — lewa łydka
62. Right foot — prawa stopa
63. Left foot — lewa stopa
64. Colon — okrężnica





10 BAJTKOWA LISTA PRZEBOJÓW 3/87

Trzecie tegoroczne notowanie Bajtkowej Listy Przebojów i trzy najnowsze gry na Atari — to wynik rosnącej popularności tego mikrokomputera. Ogółem nadesłano 5064 propozycji, głosowano na 613 gier.

			AMSTRAD	ATARI	COMMODORE	SPECTRUM
1	BOULDER DASH	↑	x		x	x
2	SPY Vs SPY II	↑		x	x	x
3	CAULDRON	↑	x		x	x
4	FIRE LORD	↓	x		x	x
5	TIGERS IN THE SNOW	!	x		x	x
6	BEACH HEAD II	↑	x	x	x	x
7	GREAT ESCAPE	↑			x	x
8	URIDIUM	↓	x		x	x
9	SEVEN CITIES OF GOLD	!		x	x	
10	SPY Vs SPY	↓		x	x	x

Nagrodę — klawiaturę muzyczną do Commodore 64 — otrzymuje Adam Szukalski z Warszawy. Nagrodę ufundowała f-ma Electronics Export

Sławek



NADAWCA: Ewa Rutkowska, 66-130 BOJADŁA, ul. Sulechowska 35.



BEACH HEAD

Jako dowódca grupy desantowej masz za zadanie uchwycić przyczółek. Na silnie ufortyfikowanym wrogim brzegu morskim musisz wysadzić desant i umocnić się stwarzając bazę do dalszego ataku w głąb lądu. Dowództwo armii liczy na Ciebie!

Na początku musisz podjąć decyzję co do sposobu forsowania brzegu. Prowadzą tam dwie drogi: łatwa przez szerokie wejście do zatoki i trudna przez wąską cieśninę. Jeżeli jednak nie zdążysz szybko przepłynąć przez szerokie wejście, to straż nabrzeżna zauważy Cię i przegrodzi wejście. Wtedy trzeba płynąć przez naszpikowaną minami cieśninę, której bronią nabrzeżne wyrzutnie torped.

Gdy znajdziesz się w zatoce, musisz stoczyć walkę z flotą nieprzyjaciela. Masz do dyspozycji jedno podwójne działo 40 mm. Z jego pomocą należy zestrzelić atakujące wrogie samoloty. Ilość ich zależy od wybranej drogi do zatoki. Po trudnym przejściu przez cieśninę samolotów jest dwa razy mniej, niż po łatwym dojściu przez szerokie wejście. Każdy nie trafiony samolot zbombarduje Twoje statki powodując dwa uszkodzenia. Statek uszkodzony dwadzieścia razy idzie na dno.

Po zestrzeleniu wszystkich samolotów czeka Cię zatopienie pięciu wrogich okrętów. Strzelasz działem 250 mm. Specjalna aparatura informuje jak daleko od celu pada pocisk. Jeżeli będzie to np. 400 jardów za blisko, to należy podnieść lufę o 2 stopnie (400/200).

Teraz masz już dostęp do brzegu i możesz wysadzić kolumnę czołgów. Trzeba sforsować nimi niebezpieczną, najeżoną minami, bunkrami i kłębami drutów kolczastych drogę, na końcu której znajduje się główny punkt obrony wybrzeża — wielki bunkier z jedenasto-osobową załogą. Dziesięciu z nich pilnuje otworów strzelniczych, a jedenasty odpowiada za działo umieszczone na szczycie bunkra. Dysponujesz czołgowym działem 114 mm, którym możesz niszczyć po drodze pojedyncze strzelnice i wrogie czołgi, lecz uważaj na słupki i murki.

Po szczęśliwym dotarciu do bunkra musisz strzelać w kolejno otwierające się otwory strzelnicze. Nie jest to proste, a gorliwa obsługa działa powoli, lecz nieubłaganie odwraca lufę w Twoją stronę i w końcu strzela. Jednym czołgiem na pewno nie zestrzelisz wszystkich okienek. Wtedy trzeba prowadzić w stronę bunkra następny czołg. Lecz uwaga — droga będzie trudniejsza! Gdy trafisz w dziesiąte okienko, ostatni obrońca niszczy działo i poddaje się wysuwając białą flagę.

Zdobyłeś brzeg!

Gra ma ciekawą, staranną grafikę i bogatą oprawę dźwiękową. Daje możliwość sterowania wszystkimi rodzajami joysticków jak i definiowania przycisków. Przechowuje też w pamięci pięć najlepszych wyników i imiona dowódców desantu.

Komputer: Atari 800XU130XE, Commodore 64/128, ZX Spectrum 48K/+
Producent: US Gold

(mp)

MUZYCZNY KOMPUTER

Adam Szukalski, lat 12, zamieszkały Warszawa, ulica H. Junkiewicza 2 m.74, uczeń VI klasy Szkoły Podstawowej nr 114 — nagrodzony w konkursie Bajkowej Listy Przebojów.

Zainteresowania: informatyka, matematyka, muzyka elektroniczna.

Sądząc po twoich zainteresowaniach komputerem, który posiadasz jest Commodore?

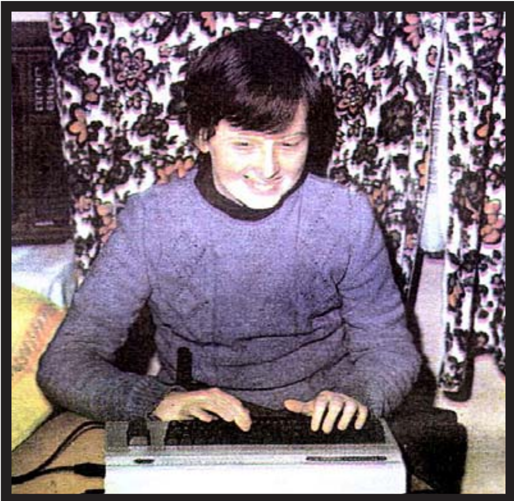
— Tak. Commodore 64 ze względu na świetny układ generujący dźwięki zaspakaja wszystkie moje potrzeby.

— Z jakich programów muzycznych najczęściej korzystasz?

— Najbardziej przyzwyczaiłem się do programu MUSIC COMPOSER.

— Myślę, że nagroda w postaci klawiatury muzycznej do C-64 napewno przyda ci się do komponowania własnych utworów.

— Marzyłem o niej od dawna.





TYGRYSY W ŚNIEGU

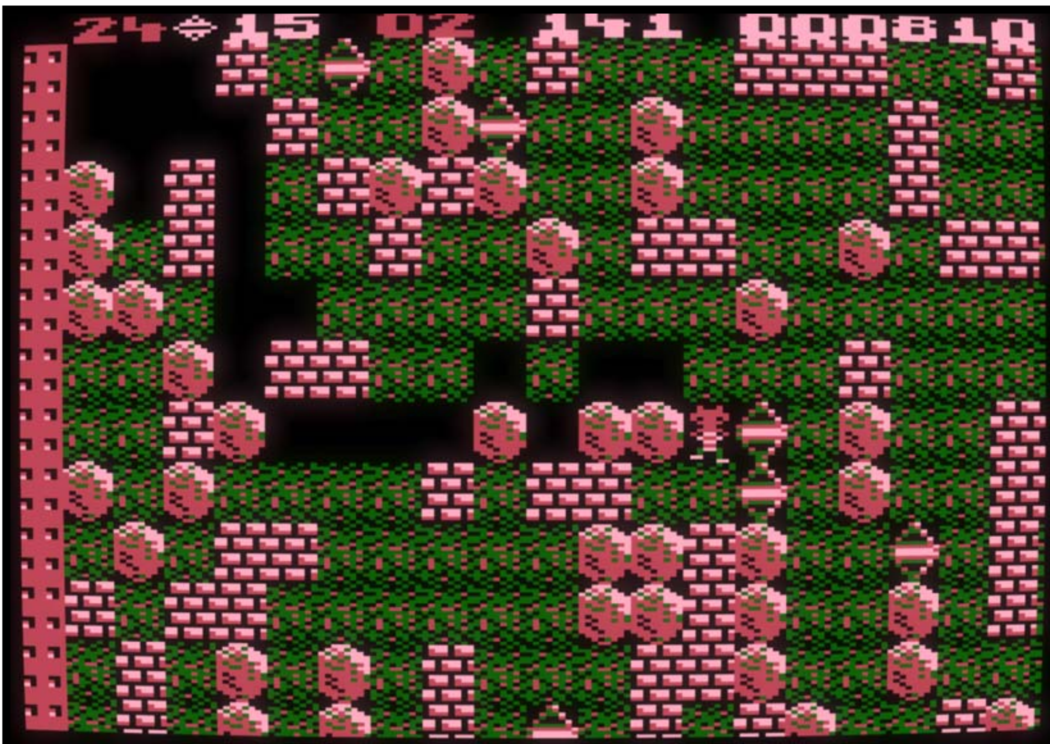
Było ciemno i ponuro. Przez warstwę niskich chmur nie przebił się ani jeden promyk wschodzącego słońca. Korony drzew spowijała gęsta mgła. Wstawał nowy dzień — 16 grudnia 1944 roku. Nagle w spokój i ciszę poranka wdarł się grzmot dział i huk setek silników. Na nie spodziewających się niczego żołnierzy czterech amerykańskich dywizji runęła lawina ognia i stali. Trzy niemieckie armie pancerne rozpoczęły ofensywę.

W tym okresie wojska alianckie kończyły już wyzwalanie Francji i szykowały się do wkroczenia na terytorium Niemiec. Nikt nie przypuszczał, że przez góry, lasy i bezdroża Ardenów można przeprowadzić czołgi — podstawowy środek walki. I chociaż już dwukrotnie (podczas wojny prusko-francuskiej i I wojny światowej) Niemcy dokonywali ofensywy w Ardenach, a może właśnie dlatego, w rejonie tym pozostawiono tylko kilka słabych i niedoświadczonych jednostek. Niemcy wykorzystali to bezlitośnie. Przerwali front i skierowali się w stronę Mozy. Alianci jednak szybko ściągnęli posiłki i zatrzymali ich. Wyróżniła się przy tym amerykańska 101 DPDes (wsławiona już walkami pod Arnhem), która została okrążona w Bastogne, ważnym węźle komunikacyjnym. Niemałą rolę w powstrzymaniu niemieckiego natarcia odegrało także lotnictwo, które sparaliżowało dowóz zaopatrzenia dla walczących oddziałów. Tym niemniej aliancka ofensywa na Niemcy została opóźniona o 6 tygodni.

Zapytacie, co to ma wspólnego z komputerami. Otóż każdy może spróbować naprawić błędy dowództwa an-glo-amerykańskiego korzystając z gry symulacyjnej „Tigers in the Snow". Gra przeznaczona jest dla dwóch lub jednego gracza (można wtedy wybrać wojska, którymi się kieruje). Symulowane są praktycznie wszystkie elementy pola walki, m.in. wpływ pogody na działalność lotnictwa i przez to na zaopatrzenie wojsk, czas potrzebny do właściwego przygotowania obrony lub ataku itd. Grający ma możliwość dowolnego wyboru jednostek atakujących i atakowanych, wsparcia artylerii oraz stosowanej taktyki walki (np. w obronie: kontratak, utrzymanie pozycji, wycofywanie, działania opóźniające). Instrukcja do gry liczy ponad 20 stron i formą przypomina instrukcje do gier firmy „ENCORE". Warto się jednak z nią dokładnie zapoznać, znacznie ułatwia to planowanie operacji.

Producent: Strategy Simulation Incorporated.
Komputery: Atari, Commodore, IBM

(ziew)



ROCKFORD CZYLI ŁOWCA DIAMENTÓW

Jeżeli chcesz mieć oczy czerwone z niewyspania i wpatrywania się w telewizor, a palce rąk tak obolałe od trzymania joysticka, że z trudem będziesz mógł je wyprostować — polecam Ci jedną z najbardziej zabawnych i rozwijających gier na ATARI firmy FIRST STAR SOFTWARE o dźwięcznym tytule „BOULDER DASH".

Ma ona tę zaletę, że na pewno nie wykończysz swojego joysticka. „BOULDER DASH" wymaga od gracza pewnego wysiłku intelektualnego, a nawet znajomości niektórych praw fizyki oraz umiejętności trafnego przewidywania.

Gwarantuję, że po trzech dniach i nocach grania wcale nie będziesz znudzony „BOULDER DASH'em", a wręcz przeciwnie.

W grze mogą uczestniczyć dwie osoby. Sterują one małą, niepozorną istotą o imieniu Rockford. Rockford, kiedy mu się coś nie powiedzie, mruga gniewnie oczami i niecierpliwie tupie nogą. Jeżeli nie potrafi znaleźć wyjścia z sytuacji, należy nacisnąć klawisz START, aby nie być wspólnikiem jego złości.

Zadaniem Rockforda jest szukanie diamentów, które przypadkiem albo być może z premedytacją, ktoś zostawił w ciągu jaskiń i grot. Bohater drąży tunele, zbierając znalezione na swej drodze szlachetne kamienie. Byłoby to bardzo proste, gdyby nie fakt, że ten sam osobnik, który porozrzucał diamenty, poprzedzielał je okropnymi głazami i blokami skalnymi. Sprawia to, że aby osiągnąć zamierzony cel należy włożyć w grę maksimum wysiłku.

Przed każdym następnym krokiem naprzód Rockford musi dokładnie ocenić sytuację. Jeden nieostrożny ruch i lawina głazów zasypuje naszego bohatera, przez co traci on jedno ze swoich trzech wcieleń. Chwila nieuwagi, bądź ułamek sekundy opóźnienia powodują, że kamienie spadają na drogę, po której się porusza.

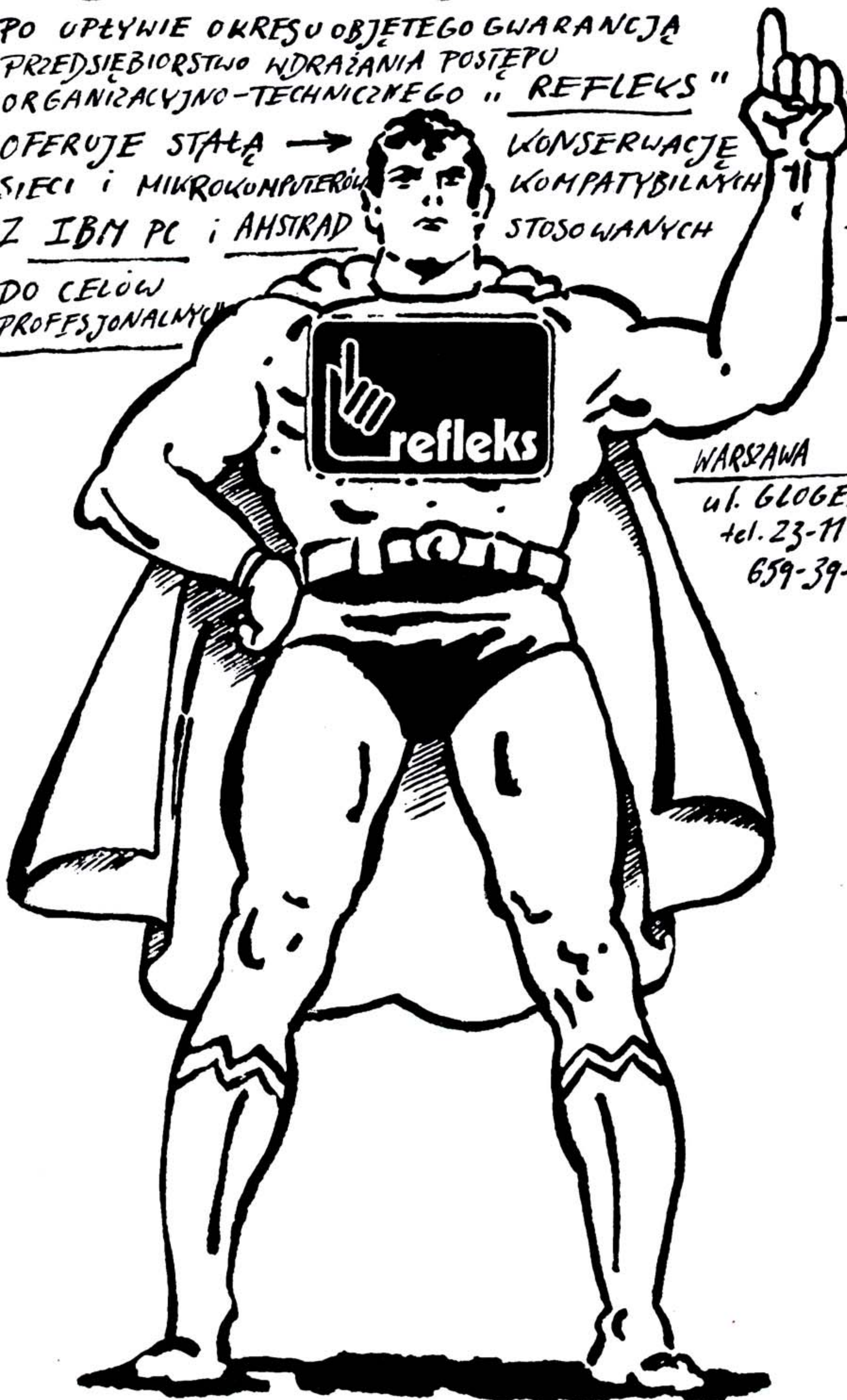
Nie jest to jedyne utrudnienie. Niezwyciężony i nieokiełznany Rockford spotyka podczas penetracji jaskiń kilku dodatkowych nieprzyjaciół. Są to: motyle, które w zetknięciu ze spadającymi głazami zamieniają się w diamenty; tajemnicze latające bomby-pułapki; bulgoczące źródła lawy oraz inne osobliwości.

Kiedy Rockford zbierze ustaloną ilość diamentów sekretne drzwi otwierają swoje podwoje i następuje przejście do następnej jaskini. Jest ich do zwiedzenia szesnaście. Rockford w każdej grocie może wybrać jeden z pięciu możliwych poziomów, na którym chce przebywać. Przejście w ramach jaskini z poziomu niższego na wyższy wiąże się z uzyskaniem premii w postaci dodatkowego Rockforda, ale również z większą ilością diamentów do zebrania, w krótszym czasie.

Teraz muszę już iść, bo jestem właśnie na dobrej drodze do sukcesu. Już druga jaskinia, a siedzę dopiero czwarty dzień.

Komputery: Atari 800 XL/130XE, Commodore 64
Producent: First Star Software

(mp)


PC XT/AT
AMSTRAD
SUPERSERWIS
PO UPŁYWIE OKRESU OBJĘTEGO GWARANCJĄ
PRZEDSIĘBIORSTWO WDRAŻANIA POSTĘPU
ORGANIZACYJNO-TECHNICZNEGO „REFLEKS" Sp. z o.o.
OFERUJE STAŁĄ → KONSERWACJĘ
SIECI i MIKROKOMPUTERÓW KOMPATYBILNYCH
Z IBM PC i AMSTRAD STOSOWANYCH
DO CELÓW PROFFESJONALNYCH
refleks
WARSZAWA
ul. GLOGERA 1
tel. 23-11-55
659-39-22

ATARI
ZX SPECTRUM

INSTRUKCJE • OPISY
LITERATURA
KATALOGI - GRATIS
SZKOŁY I KLUBY - zniżka
WYSYŁKA NA CAŁY KRAJ
WYPOŻYCZALNIA PROGRAMÓW

D.H. „SEZAM" lip., g. 16-19
00-849 WARSZAWA UPT 66
skr. p. 14.

ATCOM

```
Programy, literatura
ATARI,.C16
Najtaniej w Polsce!
P-16, 15-057 Białystok 24
tel. 419-397
```

JAK REKLAMOWAĆ SIĘ W BAJTKU?

Reklamy przyjmuje Młodzieżowa Agencja Wydawnicza (Redakcja Wydawnictw Poradniczych i Reklamy), 04–028 Warszawa, Al. Stanów Zjednoczonych 53, pokój 313. Tel. 10-56-82.

Cena ogłoszeń: 200 zł za 1 cm plus dodatki za kolor

EKLOR SOFTWARE

OFERUJE DO SPRZEDAŻY PROGRAMY DO ZX SPECTRUM:

Kasetę nr 1 cena 890,–
opis Bajtek 7/86

Kasetę nr 2 cena 1390,– .

Kasetę nr 3 cena 1490.–
- Cataloger – automatyczne katalogowanie programów
- Biblioteka – baza danych dla posiadaczy wielu programów
- TV – test – tester telewizora i monitora
- ZX – test – pełen test ZX Spectrum
- Niespodzianka – ?

Kasetę nr 4 cena 1490,.–
- Mutorere – gra logiczna
- L – gra logiczna
- Bridge – polskojęzyczna wersja Bridge 2
- Totolotkownia – komputerowa analiza gier losowych
- Edytor EL – pierwszy polski edytor BASIC u

Wszystkie programy w języku polskim pisane specjalnie dla Elkor Software. Zamówienia kierować na kartkach pocztowych: ELKOR Software, 60-120 Poznań 7, skr. pocz. 24. Wysyłka za zaliczeniem pocztowym. Nie przyjmujemy przelewów pocztowych.

D-23

BIURO USŁUG KOMPUTEROWYCH

„BONUS"

- PROGRAMY
- LITERATURA

— ATARI —
— AMSTRAD —
— ZX SPECTRUM —

04-111 Warszawa, ul. Grochowska 207, tel. 100-061 wewn. 244, w godz. 16.00 – 19.00.

D-188

–AMSTRAD CPC–

- programy użytkowe, gry poczta
- programy na zamówienie

INFORMAX 02-791 Warszawa, ul. Meander 21 m. 20

D-15

ATAROWCY,

wymieńmy programy (Dyski). Paweł Ciągło, Mokotowska 21 /204, 77-310 Debrzno.

G-4

KASETY KOMPUTEROWE Firma JACEK KOŃSKI 05-815 Opacz k/Warszawy ul. Centralna 10 C przy stacji kolejki WKD.

D-194



PROGRAMY DO KOMPUTERÓW

ZX 81, ZX SPECTRUM
ATARI 800XL

CENA PROGRAMÓW
Z KASETĄ
880, – 2.400 ZŁ

Katowice 40-337 skr. poczt. 2788

G-66

ZANIM KUPISZ KOMPUTER ZADZWOŃ

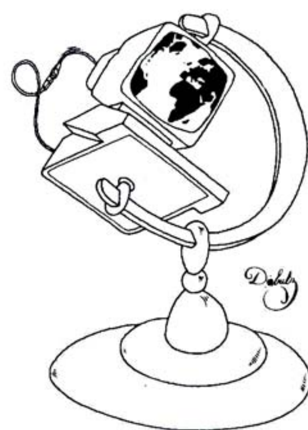
KUPIŁEŚ KOMPUTER ZADZWOŃ

28-01-76 **eurobit** **28-01-76**

PORADY:
— WYBÓR SPRZĘTU
— ZASTOSOWANIA
OPROGRAMOWANIE
INSTRUKCJE
POŚREDNICTWO
ZLECENIA

ATARI
COMMODORE C-16, 116,4
AMSTRAD
VIC20, C-64
SHARP
MSX
IBM

WYSTAWIAMY RACHUNKI DLA INSTYTUCJI
Al. Ujazdowskie 18 m. 14 Warszawa w g. 9-18



SYSTEM Sp z o. o.



CENTRUM MINIKOMPUTEROWE

Prowadzi: jako jedyne w Warszawie salon sprzedaży minikomputerów

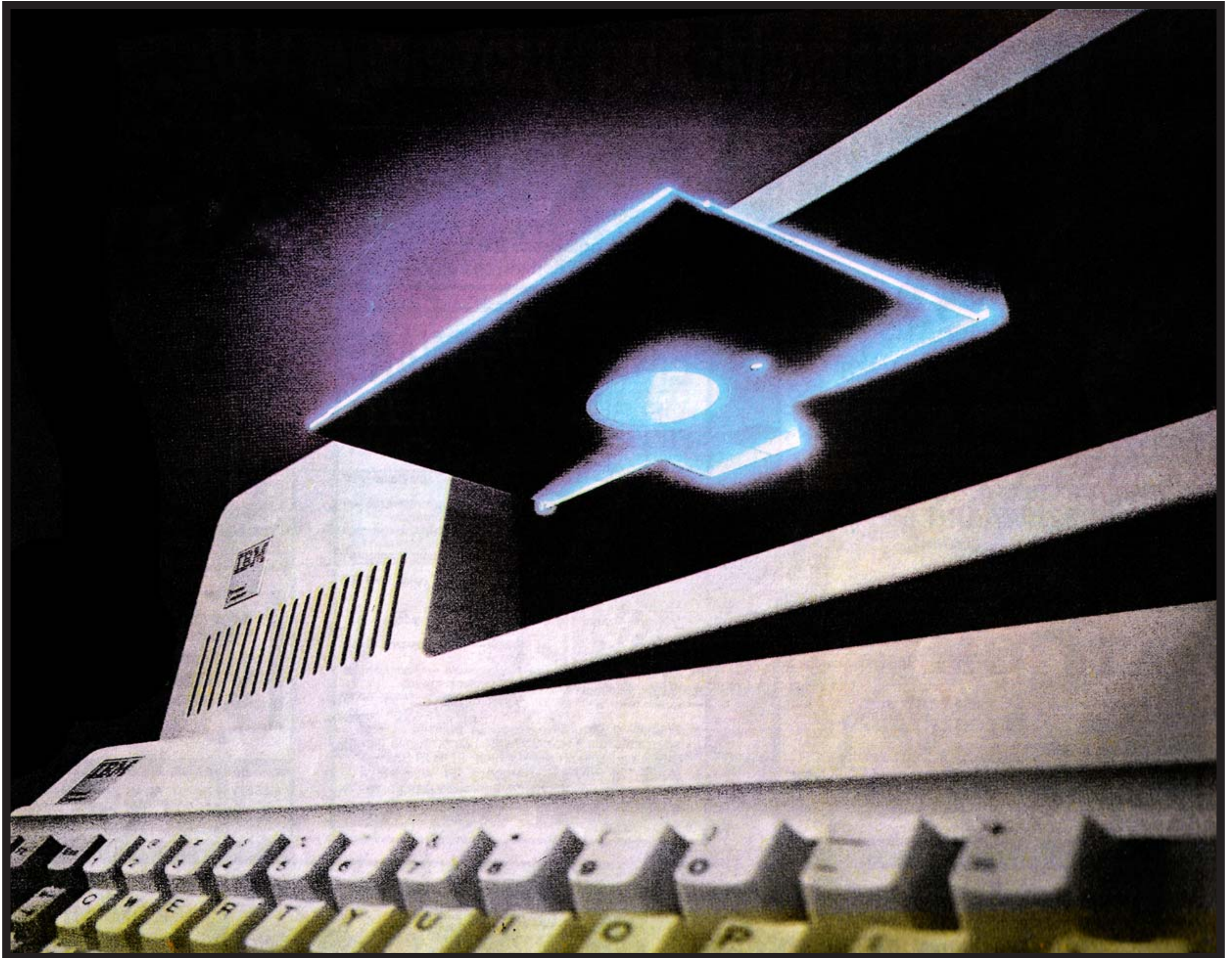
OFERUJE:

przedsiębiorstwom oraz osobom prywatnym po konkurencyjnych cenach:

* duży wybór konfiguracji sprzętu kompatybilnego XT/AT
* bogaty zestaw drukarek
* minikomputery serii Amstrad – Schneider
* inne (ATARI, COMMODORE, SPECTRUM)

Przedsiębiorstwo prowadzi również SKUP sprzętu Warszawa, ul. Wolska róg Młynarskiej (przejście podziemne – obok Domów Centrum Wola) tel. 32-80-93 godz. 11-19.

K-2



NASTĘPNY KROK
cz. 1

Zacznijmy od kilku aneksów do hasła tej kolumny. Programować rzeczywiście może każdy, ale ... Po pierwsze, nie każdy musi — można korzystać z komputerów nie umiejąc programować. Po drugie, pewne nieskomplikowane zadania można wykonać programując byle jak. Jeśli jednak chcemy pisać programy rozwiązujące duże, poważne zadania nie wystarczy po prostu programować. Trzeba programować dobrze. Bez wdawania się w zawiłe rozważania, czy jest to sztuka, czy tylko rzemiosło, możemy zauważyć, że dobremu programiście potrzebne są zarówno praktyka jak i pewien zasób wiedzy fachowej. Żeby mieć jedno i drugie potrzebny jest pewien nakład pracy. Tym artykułem rozpoczynamy cykl adresowany przede wszystkim do tych, którzy chcą taki nakład ponieść. Niestety praktykę musicie sobie zapewnić sami — jedyna metoda, którą ja znam, to pisanie i uruchamianie programów. Równocześnie chcę Was przestrzec, że „Bajtek" nie jest i nie może być podręcznikiem programowania. Dlatego nie oczekujcie systematycznego, podzielonego na kolejne lekcje wykładu. Będzie to raczej przegląd podstawowych wiadomości o programowaniu i o metodologii programowania, w miarę możności uzupełniany pożytecznymi ciekawostkami.

Nie można jednak zapominać o potrzebach tych, którzy dopiero zaczynają. Dlatego będziemy pisać również o rzeczach zupełnie podstawowych, co z kolei nie jest zapowiedzią kursu BASIC-a. Umiejętność programowania i znajomość konkretnego języka programowania to dwie różne rzeczy. My będziemy się zajmować tą pierwszą — językom „Bajtek" poświęcił już sporo miejsca.

Dziś zaczniemy nie od programowania, lecz od uporządkowania wiadomości, które przy programowaniu mogą się przydać. Bardzo często używany w informatyce jest szesnastko-wy system zapisu liczb, nazywany również hexadecymalnym (ang. hexade-cimal). Jest to system pozycyjny, działający dokładnie tak samo jak systemy: dziesiętny, ósemkowy, czy dwójkowy, opisane w numerze 12/86 „Bajtka". Podstawą systemu jest 16, do zapisu liczb potrzebujemy szesnastu znaczków, czyli cyfr systemu szesnastkowego (analogicznie do dziesięciu znaczków: 0,1,2.....9, czyli dziesięciu cyfr systemu dziesiętnego). Poradzimy sobie biorąc cyfry dziesiętne, a jako brakujące sześć wykorzystamy litery A,B,C,D,E,F. Odpowiadają one kolejnym dziesiętnym wartościom: 10,11,12,13,14,15. Tam gdzie liczby szesnastkowe mogą się mylić, np. z dziesiętnymi, zaznaczamy je pisząc na końcu literę H, np. 197H.

Kilka przykładów liczb szesnastkowych i ich wartości dziesiętnych:

1AH = 16^0 x A + 16^1 x 1 = 1 x 10 + 16 x 1 = 26

234H = 16^0 x 4 + 16^1 x3 +16^2 x2 = 1x4 + 16x3 + 256 x2 = 564

FFFH = 16^0 x F + 16^1 x F + 16^2 x F = 1 x 15 + 16 x 15 + 256 x 15 = 4095

Z powyższych przykładów widać od razu jak przeliczyć liczbę szesnastkową na dziesiętną. Jak zrobić to w drugą stronę napiszę za miesiąc, bo może spróbujecie wymyśleć to sami. System

szesnastkowy zyskał popularność dzięki ogromnej łatwości przeliczania między nim a systemem binarnym. Dwójkowe reprezentacje wartości cyfr szesnastkowych są następujące:*)

0 = 0000	4 = 0100	8 = 1000	C = 1100
1 = 0001	5 = 0101	9 = 1001	D = 1101
2 = 0010	6 = 0110	A = 1010	E = 1110
3 = 0011	7 = 0111	B = 1011	F = 1111

Widać, że każdej cyfrze odpowiada czterocyfrowa liczba dwójkowa, więc' konwersję wykonujemy mechanicznie zastępując kolejne cyfry odpowiednią czwórką bitów, np. E1A9 ma wartość binarną 111000011010. Oczywiście konwersja działa w obie strony, .< Liczbę dwójkową dzielimy na czwórki bitów, zaczynając od prawej strony,- z lewej można dopisać tyle wiodących zer, ile brakuje do pełnej czwórki. Przykład:

1101011101 (czyli 0011 0101 1101) = 35D

Zauważmy, że zawartość jednego bajtu (8 bitów) można zapisać przy pomocy dwucyfrowej liczby szesnast-kowej o wartości od 00 do FF (dziesiętnie jest to od 0 do 255). Ten sposób reprezentacji jest czasami wygodniejszy od dziesiętnego, bo łatwo zobaczyć, które bity mają wartość 0, a które 1. Piszę o tym, gdyż czasami w programach chcemy używać pewnych szczególnych komórek pamięci, których zawartości nie należy traktować jako zwykłych liczb, lecz jako ciągi bitów, z których każdy niesie inną informację. Przykładem może być zawartość zmiennej ST (STATUS), lub rejestrów generatora dźwięku na Commo-dore C-64. I chociaż najczęściej sztuki z bitami robi się przy programowaniu w assemblerze, to np. na C-64, BASIC i PASCAL pozwalają czytać zawartość różnych rejestrów (instrukcja PEEK) i zmieniać ją (POKE). Szczegóły techniczne trzeba znaleźć w odpowiedniej dokumentacji, a autorzy dokumentacji często stosują zamiennie system dziesiętny, dwójkowy i szesnastkowy, więc mimo pozornie teoretycznego wyglądu, nasze rozważania mogą się przydać w praktyce.

Wszystko to pięknie, powie ktoś, ale mój BASIC „zna" tylko liczby dziesiętne, jak w takim razie dowiedzieć się czy określony bit ma wartość jeden? Mając do dyspozycji tylko operacje na liczbach dziesiętnych musimy „oszukać" komputer, czyli potraktować interesujący nas bajt (np. pobrany przy pomocy PEEK) jako liczbę dziesiętną. Jak operując na liczbach dziesiętnych dobrać się do pojedynczych bitów?

Przypomnijmy, że operacje logiczne AND i OR, na ogół używane do budowy złożonych warunków logicznych, mogą pracować również jako operacje na liczbach dziesiętnych. Ich działanie jest wtedy następujące: obie liczby zapisane są w postaci dwójkowej, jako ciągi bitów równej długości, bity do operacji pobierane są parami, po jednym z każdego argumentu, wynik dla każdej pary zależy tylko od jej wartości — nie ma związku z resztą argumentu. Dla operacji AND wynik jednej pary jest równy 1 wtedy i tylko wtedy, gdy oba bity są 1 (w pozostałych przypadkach 0). Dla OR wynik jest 0 wtedy i tylko wtedy, gdy oba bity są 0 (w pozostałych przypadkach 1). Znowu przykład:

AND	100111001	OR	0010001
	<u>010101111</u>		<u>11000100</u>
	000101001		11010101

Zauważamy, że jeśli X — dowolna wartość binarna (0 lub 1), to

AND	XXXXXXX	OR	XXXXXX
maska	00010000	mask	00010000
	000X0000		XXX1XXX

czyli zastosowanie odpowiedniej maski i operacji AND przenosi do wyniku wartość tylko jednego, wybranego bitu. Pozostałe bity na pewno są równe 0, więc wartość tego wybranego łatwo ustalić, sprawdzając czy cały wynik jest równy 0. Operacja OR pozwala wstawić jedynkę, ustawioną w wybranym bicie maski, do bajtu bez zmiany wartości pozostałych bitów. Oczywiście kilka kolejnych OR i AND pozwoli stworzyć każdą potrzebną kombinację zer i jedynek. Można również używać masek mających kilka jedynek.

Jeszcze tylko musimy zbudować maskę, posługując się liczbami dziesiętnymi. Przypomnijmy sobie system binarny: na kolejnych pozycjach (bitach) stoją kolejne potęgi dwójki, a każda z nich ma znaną wartość dziesiętną. Popatrzmy na bajt: Bity:

	b7	b6	b5	b4	b3	b2	b1	b0
Potęgi:	27	26	25	24	23	32	21	20
Dziesiętne:	128	64	32	16	8	4	2	1

Nie trudno zgadnąć, że maską, która wytnie bit b5 jest po prostu liczba 32,. itd. Przykłady:

ustawić b6 na 1 w bajcie o adresie 100:

POKE 100, PEEK(100) OR 64

sprawdzić, czy bit b3 w zmiennej A% jest = 0:

IF (A% AND 8) = 0 PRINT „b3 = 0"

sprawdzić, czy bajt 5537 ma bity b1 i b5 równe 1:

IF (PEEK(5537) AND 34) = 34 PRINT „b1 = 1 i b5 = 1"

(oczywiście 34 = 32 + 2, maska zawiera dwie jedynki: 00100010).

UWAGA: przy wszystkich operacjach bitowych należy posługiwać się liczbami (stałymi i zmiennymi) całkowitymi, czyli typu INTEGER. Zmienne rzeczywiste zapisywane są w maszynie w całkiem inny sposób niż omówiony przez nas zwykły system binarny.

Na zakończenie pytanie: jak sprawdzić, czy wartość zmiennej całkowitej jest parzysta? Jak zrobić to samo nie korzystając z operacji AND? Odpowiedź oczywiście za miesiąc.

Andrzej Pilaszek

*) Jeśli interesuje Cię dodawanie w arytmetyce binarnej możesz je prześledzić na podstawie tej tabelki. Kolejna w kolumnie liczba dwójkowa powstaje przez dodanie 1 do poprzedniej.

NAPRAWDĘ NIE ŚWIĘCI GARNKI LEPIĄ

Propozycja stworzenia własnej bazy danych („Bajtek" 8/86) przypadła do gustu części naszych Czytelników. Dziś przedstawię dwa z nadesłanych programów. Oba, mimo pewnych zastrzeżeń, mogą z powodzeniem spełniać zaplanowane funkcje i chciałbym w tym miejscu przekazać słowa uznania dla Autorów. Uwag zawartych w dalszej części proszę nie traktować jako surową krytykę, lecz raczej jako rady co jeszcze warto wziąć pod uwagę przy projektowaniu programów tego typu.

Autorem pierwszej bazy, uruchomionej na Atari 800XL jest JERZY FIOŁKA z Warszawy. Bezpośrednio po załadowaniu programu na ekranie pojawia się następujące menu:

*** **MENU** ***

1) Wprowadzanie zapisów.
2) Zmiana zawartości zapisów.
3) Podaje pierwszy napotkany zapis wg dowolnego wzorca.
4) Podaje wszystkie napotkane zapisy wg dowolnego wzorca.
5) Zapis bazy z kasety.
6) Zapis bazy na kasetę.
7) Kasowanie zapisów.
8) Zakładanie bazy. KASUJE STARĄ!!!

Wybierz pozycję

UWAGA: w każdej sytuacji „pusty" RETURN powrót do MENU.

Brakuje trochę możliwości dowiedzenia się czegoś bliższego o ogólnych zasadach pracy z programem — mogło by to być np. kilka ekranów tekstu, wyświetlanych po wybraniu (nieistniejącej w tej chwili) opcji: „Wyświetlanie skróconej instrukcji". Na szczęście po wybraniu poszczególnych punktów z menu program informuje co robić dalej. Np. w punktach 3, 4 prosi o podanie wzorca. Jak funkcjonuje wzorzec niestety trzeba zgadnąć samemu. Otóż baza jest to ciąg rekordów, z których każdy może zawierać tekst Jeśli podamy wzorzec, np. arb to program znajdzie teksty zawierające w sobie ciąg znaków arb np. Barbara, rabarbar itd.

Brak tablic tekstowych w komputerze Atari autor ominął zapisując rekordy jako jeden długi tekst, w którym rekordy rozdzielone są znakiem @ Znak ten oczywiście nie może być zawarty w treści wpisywanych rekordów. Program informuje o tym zakazie, ale niestety nie sprawdza czy użytkownik go nie złamał. Nieco inaczej wygląda baza danych na Spectrum, której autorem jest MARCIN WITKIEWICZ z Warszawy. Główne menu jest tu nieco uboższe (PRZEGLĄDANIE, WYSZUKIWANIE, ZAPISYWANIE, DOPISYWANIE, ZMIANY), ale po wybraniu każdej z funkcji otrzymujemy następne menu (każda z funkcji ma własne), pozwalające wybierać bardziej szczegółowe czynności. Wyświetlane są także ułatwiające pracę wyjaśnienia.

Każdy rekord bazy składa się ź trzech pól, nazwy pól można wybrać samemu, wyszukiwanie rekordów odbywa się przez porównywanie podanego przez użytkownika wzorca z zawartością pierwszego pola każdego rekordu. Wzorzec porównywany jest tylko z początkiem, tekstu, tzn. tekst Bajtek będzie znaleziony dla wzorców takich jak Baj, Ba, Bajtek, natomiast (w przeciwieństwie do bazy Atari) nie będzie znaleziony dla wzorców: ajt, jtek, itd. Która z metod jest lepsza — to zależy od konkretnego zastosowania. Najlepiej by było, gdyby użytkownik miał do wyboru obie.

W bazie mieści się na stałe 100 rekordów. Te, do których nie wpisaliśmy danych przechowywane są jako puste. Przy zapisie danych podajemy numer rekordu, do którego mają zostać wpisane. Program wyświetla numery pustych rekordów, ale to za mało, powinien jeszcze sprawdzić, czy rekord, który wybrałem jest naprawdę pusty — brak tej kontroli może spowodować przypadkowe zniszczenie poprzednio wpisanych danych. Okazuje się jednak, że zaufanie do użytkownika jest typową cechą (przypomnijmy znak @ w bazie Atari). Następnym jego objawem jest pytanie czy baza już jest wczytana z kasety. Tego typu informacje program jest w stanie zapamiętywać sam.

I jeszcze rzecz wspólna w obu bazach: po wybraniu opcji „poprawianie rekordu" całą zawartość trzeba wpisać na nowo. Dużo wygodniej jest jeśli starą zawartość wyświetlimy na ekranie a użytkownik może przesuwać się po niej kursorem i nanosić poprawki — po co pisać na nowo 100 znaków, jeśli tylko jeden jest zły, a pozostałe dobre.

Na zakończenie raz jeszcze chciałbym podkreślić, że rozwiązania, które sugeruję znajdują się zwykle w poważnych bazach danych, tworzonych przez zespoły zawodowych programistów. Dlatego nie można mieć za złe komuś kto samodzielnie stworzył pierwszą w życiu bazę, że nie wszystko wziął pod uwagę.

Programy, które otrzymałem wykonują zadanie, do którego zostały przeznaczone i to jest najważniejsze-Dziękuję za ich nadesłanie i życzę Autorom dalszych sukcesów.

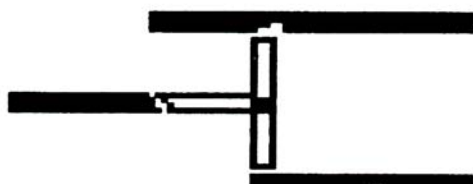
a.p.

DREWNIANY KOMPUTER

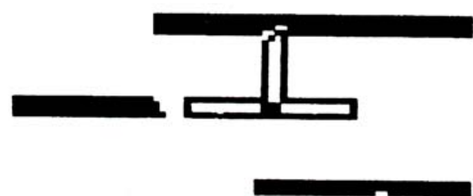
Spróbujmy zrozumieć, jak można praktycznie zrealizować przy pomocy urządzenia przyjmującego tylko dwa stany, tak abstrakcyjne czynności jak operacje matematyczne. Czyli, po prostu, jak się to robi, że to działa!

Jak można sobie to wyobrazić? Spróbuję w tym pomóc. Aby to wszystko było nam bardziej bliskie, stworzymy sobie nie małe krzemowe urządzenie, w którym płyną maleńkie elektrony, ale duże drewniane, o nazwie flip-flop (rys. 2), w którym rolę tych ostatnich spełniają kulki (rys. 1).

Przyjrzyjmy się teraz bliżej działaniu flip-flopa. Gdy wahadełko jest w położeniu poziomym (rys. 1a), umówmy się, że jest to jego poziom niski, czyli stan zero, gdy jest w położeniu pionowym (rys. 1 b), jest to jego poziom wysoki, czyli stan jeden. Przelot kulki przez flip-flopa zmienia jego stan i ustawia z zera na jeden lub na odwrót. Ustawienie wszystkich flip-flopów w urządzeniu określa jego stan. Ciekawi mnie czy wszyscy wierzą, że tak skonstruowane urządzenie może coś policzyć? Zaraz to jednak udowodnię. Zamierzam wykonać dość prostą operację dodawania 3 + 7. Można tu się uśmiechnąć i pomyśleć, co to za komputer, który nic bardziej skomplikowanego nie potrafi. Ale prawdziwy komputer posiada bardzo dużo węzłów, a ten tylko cztery. Flip-flop przyjmują dwa położenia, czyli zaprezentowana konstrukcja, jeśli będzie działała to w systemie dwójkowym. (Przypomnij sobie artykuł z n-ru 12'86 Bajtka — „Dwa palce komputera"). Dla ułatwienia przyjmijmy, że wskaźniki pisane małymi cyframi w indeksie przy liczbie oznaczają w jakim systemie jest ta liczba napisana.



Rys. 1a



Rys. 1b

Zatem komputer start. Jest on w stanie „zerowym" (rys. 2a). Siadamy przed klawiaturą i wciskamy liczbę trzy, tzn. wrzucamy dwie kulki do kanałów A i B. Na rysunku 2b widać, że tak jest rzeczywiście. Dwa pierwsze sąsiednie flip-flopy wychyliły się co daje liczbę trzy 112=310 na naszym ekranie. Następnie wciskamy liczbę siedem, wrzucają trzy kulki do kanałów A, B, C (1112=710). Po wykonaniu tej operacji stan wysoki przyjmują flip-flopy w kanałach B i D (rys. 2c). Otrzymaliśmy stan komputera równy 10102 = 1010. A więc zgadza się!

Dzwonek przed urządzeniem służy do sygnalizacji przekroczenia zakresu pojemności urządzenia, czyli suma nie może przekraczać 15 (11112=1510).

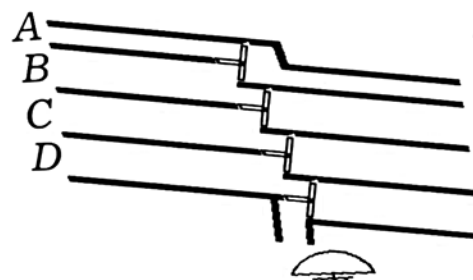
Na rys. 3a i b przedstawiony jest inny sposób realizacji flip-flopów. Kierunek wychylenia wahadełka sygnalizuje stan flip-flopa.

Ktoś teraz może powiedzieć: no i dobrze, ta niesamowita konstrukcja nawet coś policzyła, ale ostatecznie to bardziej przypomina kalkulator (i to bynajmniej nie kieszonkowy) niż komputer.

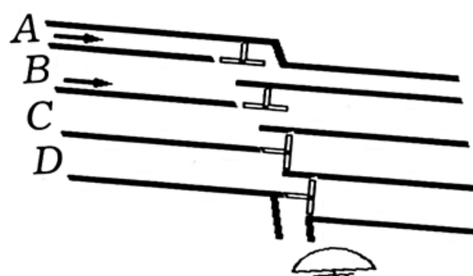
Właściwie to trzeba się z powyższym zgodzić, ale jest takie małe „ale".

Podany przykład dodawania binarnego (dwójkowego) po uważniejszym przeczytaniu i zrozumieniu pokazuje mechanizm arytmetyki dwójkowej, tzn. przesuwanie się stanów zero-jedynkowych tu w przypadku dodawania. Taka sama istnieje w komputerach, i tu czas na drugą niespodziankę. Mikroprocesor, tzn. ten mózg komputera właściwie niewiele więcej potrafi. Dodatkowe funkcje to: umiejętność przesyłania konkretnych stanów zer i jedynek między różnymi częściami (węzłami) komputera, wykonywanie oprócz operacji arytmetycznych operacji logicznych na stanach komputera, tzn. sumy i iloczynu logicznego. Tym jednak zajmiemy się w następnych odcinkach.

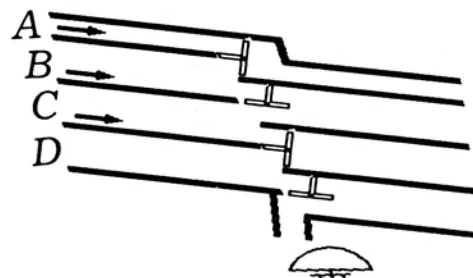
Krzysztof Czernek



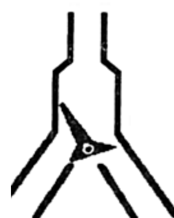
Rys. 2a



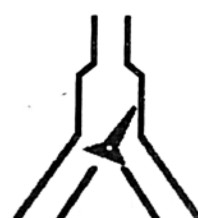
Rys. 2b



Rys. 2c

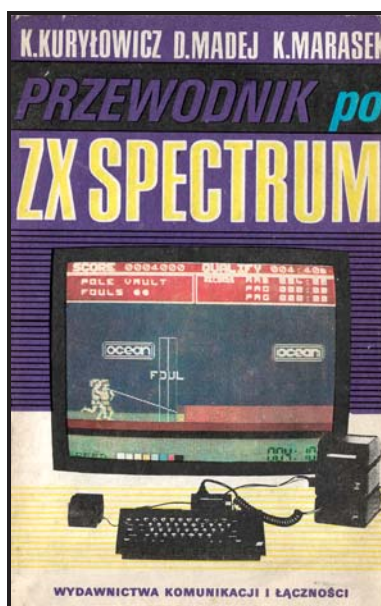


Rys. 3a



Rys. 3b

PRZEWODNIK PO ZX SPECTRUM



Doczekaliśmy się wreszcie pierwszej, z prawdziwego zdarzenia książki poświęconej ZX Spectrum. Jest nią wydany przez Wydawnictwa Komunikacji i Łączności „Przewodnik po ZX Spectrum" Krzysztofa Kurytowicza, Dariusza Madeja i Krzysztofa Maraska. Dwóch pierwszych autorów nasi czytelnicy pamiętają jeszcze z pierwszych numerów „starego" Bajtka.

Książka zawiera 9 rozdziałów adresowanych do użytkowników o różnym stopniu wtajemniczenia. Może ona zainteresować nie tylko tych, którzy po raz pierwszy usiądą przy klawiaturze, ale również bardziej doświadczonych programistów poszukujących szczegółowych wiadomości na temat systemu.

Pierwsze strony mają na celu zaznajomienie z podstawowymi zasadami obsługi Spectrum i jego budową. Zawierają też wiele cennych wskazówek praktycznych. Dalsza lektura wprowadza w świat instrukcji BASIC-a i reguł posługiwania się nim. Opis przeplatany jest przystępnymi objaśnieniami stosowanych terminów informatycznych oraz dużą ilością przykładów. Czytelnika mającego większą praktykę zainteresują z pewnością szczegółowe informacje dotyczące struktury pamięci RAM oraz przeznaczenia zmiennych systemowych.

Bardzo starannie opracowany jest rozdział poświęcony grafice. Autorzy słusznie podkreślają jej duże znaczenie nie tylko przy pisaniu gier. Omówiono sposoby czytelnej prezentacji wyników obliczeń i tworzenia wykresów, ilustrując je konkretnymi przykładami. Wiele uwagi poświęcono budowaniu własnej grafiki podając gotowe procedury rozszerzające standardowy generator znaków. Nie brak również danych na temat animacji obrazu. Szkoda natomiast, że autorzy nie napisali więcej o współpracy z drukarką. Półtorastronicowa wzmianka to stanowczo za mało.

Kolejny rozdział dotyczy programowania w asemblerze. Zrozumienie poruszanych tu problemów jest znacznie trudniejsze, ale za to pozwala głębiej sięgnąć do oprogramowania systemowego. Można tu dowiedzieć się więcej o zmiennych systemowych, wykorzystywaniu procedur z pamięci ROM i współpracy z urządzeniami zewnętrznymi. Także piraci znajdą coś dla siebie, bowiem autorzy przeznaczyli parę stron na sposoby zabezpieczania programów i łamania tych zabezpieczeń.

Dalsza część książki dotyczy posługiwania się microdri-re'm i pamięcią dyskową. Ze względu na duże znaczenie stacji dysków współpracującej ze Spectrum i częste jej stosowanie (wiele placówek naukowych do dziś wykorzystuje te komputery nawet do poważnych prac naukowoba-dawczych) powinno się napisać o niej znacznie więcej.

Ostatni rozdział zajmuje się oprogramowaniem. Autorzy opisali w nim kilkanaście programów użytkowych i gier. Bardzo wartościowym materiałem są dodatki umieszczone na końcu: zestaw znaków kodu ASCII, lista rozkazów Z80, wykaz zmiennych systemowych, komunikaty błędów oraz opis złącza krawędziowego.

„Przewodnik po ZX Spectrum" jest bardzo wartościową książką opracowaną wszechstronnie, rzetelnie i fachowo. Udało się w niej uwzględnić potrzeby i zainteresowania różnych grup odbiorców. Powinna się ona znaleźć w biblioteczce każdego — obecnego lub przyszłego — użytkownika Spectrum. Radzimy więc zawczasu postarać się o jej upolowanie w księgarni, bowiem mimo stosunkowo dużego nakładu — 150000 egzemplarzy, nie będzie długo leżeć na półkach.

(jj)

Na prośbę autorów zamieszczamy mini-erratę do „Przewodnika..."

1. W schemacie blokowym na str. 21 zamieniono miejscami napisy 16KB i 32 KB RAM
2. Zamieniono miejscami kolejność przykładów użycia OVER i AT na stronach 82 i 83
3. Podano niewłaściwą długość jednobaitowej zmiennej systemowej SUBPPC na str. 278

Krzysztof Kuryłowicz, Dariusz Madej, Krzysztof Mara-sek — „Przewodnik po ZX Spectrum", Warszawa 1986, Wydawnictwa Komunikacji i Łączności. Wydanie I. Nakład 150 tys. egz. Cena 490 zł.

POZNAJ SWÓJ KOMPUTER



Są takie książki dla dzieci, po które chętnie sięgają dorośli. Jedni z zażenowaniem, gdy nikt nie widzi, inni chwalą się tym otwarcie. Kubuś Puchatek i Mały Książe przeszły już do klasyki „dziecięco-dorosłej" literatury".

Okazuje się, że dotyczy to także wydawnictw popularno-naukowych. Książkę Friedricha Haugga o zachęcającym tytule „Poznaj swój komputer" — dobre tłumaczenie z niemieckiego Krzysztofa Sołtyskiego — będą zapewne czytały z równym zapałem dzieci i ich rodzice. Poznawanie naszego komputera rozpoczynamy od zasady działania latarki elektrycznej a kończymy na... komputerach pokładowych samolotów i statków kosmicznych. I jest to naprawdę rzetelna wiedza. Amatorzy bezmyślnego wklepywania programów nie znajdą tu nic dla siebie. Najważniejszym elementem książki nie jest wcale programowanie — a to, jak działa komputer.

Dowcipne, nietrywialne przykłady (jak choćby ten o kuzynach Kaczora Donalda, którzy mieli tylko po osiem palców i dlatego musieli nauczyć się liczyć w systemie ósemkowym), barwny język, przejrzysta kompozycja całości sprawiają, że książkę czyta się lekko i z przyjemnością — choć nie jednym tchem.

Konstrukcja książki przypomina podręcznik. Do zrozumienia kolejnych jej rozdziałów potrzebna jest wiedza nabyta w poprzednich. Warto więc zatrzymać się co parę stron i poukładać w głowie świeżo zdobyte wiadomości. Autor pomyślał również i o tym, umieszczając w tekście 22 zadania do samodzielnego rozwiązania.

Kupując tę książkę — czy mamy 14, czy też 10 lat — zapewniamy sobie pasjonującą lekturę na wiele, wiele wieczorów. A po jej przeczytaniu z pewnością nie będziemy już szukać pod klawiaturą maleńkich krasnali w czerwonych czapeczkach. (fT))

Friedrich Hangg „Poznaj swój komputer". Z języka niemieckiego przełożył Krzysztof Sołtyski. Wydawnictwo Naukowo-Techniczne. Warszawa 1986. Nakład 100 tys. egz. Cena 400 zł.

GIEŁDA (ceny na dzień 87.01.31 r.)

		GIEŁDA BAJTKA (tys. zl)	KOMIS (tys. zl)	AUSTRIA (średnie) (öS)	FRANCJA (średnie) (FF)	RFN (średnie) (DM)	WLK. BRYT (średnie) (£)
SINCLAIR	ZX81	25–35	–	580	–	49	–
	ZX Spectrum 48 kB	90-95	–	1400	–	150–250	45–65
	ZX Spectrum Plus	120–140	140–160	1590	1350	180–300	65–75
	ZX Spectrum 128 kB	270	–	–	1990	–	120–130
	Drukarka SEIKOSHA GP50S	100–120	–	–	–	199	60–65
	Interface Kempston	6–15	20	250	300	35	6–9
	Stacja dyskietek OPUS 1 (3,5'')	–	–	1950	–	395	95–105
COMMODORE	C-64	190–200	–	3500	1900	370–449	90–110
	C-128	310–340	430	6500	2890	650–700	210–230
	C-128D	650-700	–	12000	6850	1250	390–410
	Amiga z monit. kolorowym	–	–	–	–	2900	1099
	Magnetofon 1531	35-37	40	900	350	65	25
	Stacja dyskietek 1541	200	275	4500	1950	470	110–150
	Stacja dyskietek 1570	–	–	6900	2300	490–540	160
	Drukarka MPS 801	160	195	2900	2200	199	–
	Dyskietki 51/4 (średnia jak.)	0.5–1.5	1–2	10–25	7	0.5–2.5	0.8–2
ATARI	800 XL	100–115	90–100	1500	900	140–180	60
	130 XE	200	–	2100	1400	360	150
	Stacja dyskietek 1050	180	160–200	2200	2150	370	130
	Magnetofon	25–30	30–35	800	–	69	20
	Drukarka 1029	170-200	–	1990	–	–	85
AMSTRAD	464 z mon. monochromat.	150-270	300	6500	2690	520	160
	6128 z mon. monochromat.	410	700–800	11000	3990	990	250
	6128 z mon. kolor.	485-500	1mln	15800	5290	1250	320
	PCW 8256	–	1.3 mln	–	5920	1500	340
	Dyskietki 3"	3–7	6–7	–	35	7-12	3.5–4
	Stacja dyskietek 3'' do 464	–	295	6000	260	549	140
	PC 1512 CD	1.5 mln	2.1 mln	–	5920	1189	400

SPOWALNIACZ

Brrr.....Finansowy" dobrał się do giełdy! I my, i koledzy z giełdy w „Stodole" od trzech tygodni zauważyliśmy coraz większynapływ potencjalnych kupujących na nasze giełdy oraz coraz mniejszą ilość sprzedawców. Sprzęt, peryferia i akcesoria reprezentowane są skromnie. Na nabywców czekają pojedyncze egzemplarze mikrokomputerów, a bardziej przedsiębiorczy kupcy próbują dobić targu jeszcze przed budynkiem giełdy.

W tej sytuacji oferta handlowa obejmuje głownie literaturę o programach, językach programowania, instrukcje obsługi i opisy gier, a także podręczniki dla mniej lub bardziej zaawansowanych użytkowników. Uzupełnieniem asortymentu są interface własnej produkcji do Atari oraz interface do joysticka typu Kempston do Spectrum, najlepsze egzemplarze oparte na układzie scalonym Z80PIO firmy ZILOG.

Natomiast prawdziwą nowością i atrakcją dla początkujących zwolenników gier stała się przystawka — SPOWALNIACZ, dzięki któremu nawet najszybszą i najbardziej skomplikowaną grę można zakończyć zwycięsko.



INDYWIDUALNY BANK DANYCH

Nazywam się **Izabela Klunder**, jestem uczennicą i mam 13 lat. Posiadam mikrokomputer VZ 200 Colour. Interesuję się krótkofalarstwem i elektroniką. Z innymi użytkownikami tego typu mikrokomputera chciałabym wymienić informacje dotyczące VZ 200, jego zastosowania i możliwości wykorzystania. Mój adres: ul. Armii Wojska Polskiego 73a/17, 86-100 Świecie.

Witold Słowiński, kierownik sekcji elektr., 34 lata. Mikrokomputer COMMODORE 16 i TIMEX 2048, Da-tassette 1531 i Timex 2020. Zainteresowania: elektronika, elektrotechnika, fotografia. Oczekuję informacji o wykorzystaniu mikrokomputerów w pracy zawodowej oraz proponuję wymianę programów i wypożyczanie literatury. Adres: ul. SasankowaI/45 20-537 Lublin.

Mateusz Znaiński, 13 lat. Mikrokomputer Colour Genie EG 2000. Zależy mi na kontakcie z innymi posiadaczami tego typu mikrokomputera, a głównie na wymianie oprogramowania. Adres: ul. 22 Lipca 3/1, 45-093 Opole.

Mirosław Grzybowski, uczeń, 17 lat. Mikrokomputer Schneider CPC 464. Oprogramowanie: program pomocniczy do sprawdzania znajomości kodeksu drogowego. Zainteresowania: informatyka, elektronika. Wymiana programów oraz czasopism. Adres: ul. Wesoła 814, 67-100 Nowa Sól.

Andrzej Kozioł, uczeń LO, 18 lat. Mikrokomputer Amstrad 464 i 6128 oraz syntezator mowy d'k Tronics. Oprogramowanie: diagramy kolumnowe, sprite editor, grafika komputerowa... Zainteresowania : informatyka, elektronika, astrofizyka, języki programowania, literatura science-fiction. Adres: ul. Kolejowa 34/10, 72-610 Międzyzdroje.

Czesław Sakowicz, mgr inż. elektronik, 42 lata. Mikrokomputer ATARI 800 XL z firmowym magnetofonem. Oprogramowanie: problemy matematyczne, zawodowe obliczenia techniczne, programy kalkulacyjne. Zainteresowania: matematyka i jej zastosowania. Adres: ul. I Armii WP 33g/5, 84-300 Lębork.

Damian Szymaczek, student informatyki, 21 lat. Mikrokomputer ATARI 1040 ST oraz drukarka i myszka. Zainteresowania: grafika komputerowa, język C, metody nauki programowania — 12—16 lat. Wymiana programów użytkowych oraz doświadczeń w pracy z Atari 1040 ST i innymi. Adres: ul. Narutowicza 3, 42-600 Tarnowskie Góry.

Radosław Boczek, uczeń, 13 lat. Mikrokomputer ATARI 130 XE oraz drukarka Atari 1027 i stacja dysków 1050. Adres: ul. Toruńska 78/63, 88-100 Inowrocław.

Tadeusz Bartkowiak, inż. elektryk, 39 lat. Mikrokomputer ATARI 260 ST i stacja dysków SF 354 oraz myszka. Oprogramowanie: obliczanie oświetlenia stadionów sportowych i terenów otwartych, szczególnie dla telewizji kolorowej. Zainteresowania: urządzenia sportowe i elektromechanika precyzyjna. Adres: (od maja 1987) ul. Horsztyńskiego 22, 65-525 Zielona Góra.

Tomasz Siemczyk, uczeń, 11 lat. Mikrokomputer: Commodore 64, Datassette 1531, organy. Oprogramowanie: programy użytkowe i gry. Zainteresowania: technika i informatyka. Wymiana oprogramowania i doświadczeń. Adres: ul. Rajska 7, 71-476 Szczecin.

Paweł Nowak, uczeń, 17 lat. Mikrokomputer: Commodore VIC-20, magnetofon, expander pamięci. Oprogramowanie: głównie użytkowe. Zainteresowania: elektronika, informatyka, krótkofalarstwo. Chciałbym wymienić oprogramowanie oraz poszerzyć wiedzę na temat asemblera VIC-20. Adres: ul. Bałkańska 6/27, 85-167 Bydgoszcz.

Przemysław Koziarski, uczeń, 17 lat. Mikrokomputer: Commodore PC-128, stacja dysków 1571, magnetofon i drukarka. Oprogramowanie: programy użytkowe (liczące, sortujące itp.) oraz gry. Zainteresowania: fotografika i elektronika. Pragnę wymieniać oprogramowanie, literaturę oraz doświadczenia. Adres: ul. Pasieczna 4a/10, 45-087 Opole.

Andrzej Zieliński, kierownik transportu, 30 lat. Mikrokomputer SHARP MZ 731. Oprogramowanie: gry, programy edukacyjne i obliczeniowe, asembler. Wymienię oprogramowanie i doświadczenia. Adres: ul. Sobieskiego 229/10, 84-200 Wejherowo.

W BRATYSŁAWSKIM T-KLUBIE

W Czechosłowacji w mikrokomputery wyposażone są wszystkie rolnicze spółdzielnie produkcyjne. Przy pomocy komputera dokonuje się zamiany mieszkań. Komputery obecnie są również w szkołach, na wyższych uczelniach i w klubach młodzieżowych.

Odwiedziłem jeden z takich klubów, działający w Bratysławie pod patronatem czechosłowackiego Socjalistycznego Związku Młodzieży . T-Klub, taką bowiem nosi nazwę, funkcjonuje od niedawna — zaledwie od paru miesięcy. Działa jednak niezwykle sprawnie i cieszy się niemałą popularnością. Do klubu może przyjść każdy — nie tylko członek organizacji młodzieżowej — kto interesuje się mikroelektroniką. Każdy też ma swobodny dostęp do komputerów (PMD produkcji czechosłowackiej) czy video. Oprócz seansów video czy komputerowych zabaw klub stawia sobie również ambitniejsze zadania. Organizuje giełdy programów komputerowych, pokazy filmów szkoleniowych, kursy podstaw programowania dla instruktorów klubów młodzieżowych i kółek zainteresowań na terenie całej Słowacji. Prowadzi też kursy korespondencyjne wydając „przy okazji" ciekawe broszurki pisane przystępnym, zrozumiałym nawet dla laika językiem. Oto niektóre tytuły: „Programowanie dla początkujących", „Technika obliczeniowa", „Programowanie w języku BASIC", „Co umie, a czego nie ' umie komputer?". Pracownicy klubu udzielają też porad i konsultacji w takich m.in. „sferach tematycznych", jak konstrukcja, działanie i wykorzystanie osobistych mikrokomputerów, technika audiowizualna, kompleksowa automatyzacja.

T-Klub otwarty jest praktycznie codziennie, również w soboty i niedziele, od rana do późnych godzin popołudniowych. Przed południem udostępnia swe pomieszczenia — oczywiście wraz ze sprzętem — studentom Słowackiej Wyższej Szkoły Technicznej oraz uczniom bratysławskiej Średniej Szkoły Ekonomicznej, którzy tu właśnie odbywają zajęcia. W czasie wolnym od pracy i nauki klub przekształca się w forum entuzjastów i sympatyków technicznych nowości. Odbywają się tam wówczas spotkania kółek zainteresowań, prelekcje, seminaria. Członkowie klubu mają też możliwość bezpośredniej pracy z komputerem, mogą opracowywać programy oraz rozwiązywać konkretne problemy. W ten sposób tworzy się bank programów, uzupełniony bogatą literaturą fachową.

Kto przychodzi do klubu? Przede wszystkim młodzież w wieku szkolnym, choć klub otwarty jest także dla przedstawicieli starszej generacji. W trakcie jednej z wizyt w T--Klubie trafiłem na zebranie „Klubu Atari". Na sali zauważyłem również osoby w wieku „ponadmłodzieżowym". Młode natomiast jest kierownictwo klubu. Kierownik, inżynier Eduard Vachalik, „na oko" nie ma chyba 30 lat. Jest nie tylko fachowcem i sprawnym organizatorem, ale także prawdziwym pasjonatem elektroniki i działalności klubowej. Jeżeli nie ma go na terenie klubu, oznacza to, że właśnie załatwia jakieś sprawy klubowe w innej instytucji.

Z inżynierem Vachalikiem, notabene czytelnikiem „Bajtka", odbyłem kilka długich rozmów na temat działalności T-Klubu, perspektyw i planów na przyszłość. „Spodziewamy się, że niebawem dostaniemy dodatkowe pomieszczenia — mówi szef T-Klubu — pozwoli to nam rozwinąć działalność gospodarczą. Chcielibyśmy rozpocząć eksperymentalną małoseryjną produkcję unikalnych prototypów". Czy projekty takie już powstały? Okazuje się, że tak. Z początkiem roku szkolnego 1986/87 ogłoszono drugą edycję konkursu „Elektronika na co-dzień", którego organizatorem jest w tym roku właśnie T-Klub. Konkurs ten ma na celu nie tylko, cytując regulamin, „ukierunkowanie twórczej inicjatywy młodych ludzi na rozwiązywanie zadań związanych z wykorzystaniem elektroniki w działalności szkolnej i pozaszkolnej", lecz również udział w rozwiązywaniu pewnych problemów praktycznych. SZM pragnie przyczynić się, właśnie drogą konkursu, do likwidacji niedostatecznego zaopatrzenia rynku w pomoce naukowe, gry i zabawki elektroniczne, przyrządy pomiarowe, do lepszego wykorzystania elektroniki w gospodarce — oszczędności paliw, energii i ludzkiej pracy.

Konkurs odbywa się w dwóch kategoriach. Pierwsza — dla młodych konstruktorów w wieku do 35 lat — obejmuje opracowanie projektów nowych przyrządów pomiarowych, uzupełniających urządzeń elektronicznych, mających na celu poprawę użytkowych parametrów towarów rynkowych i mikrokomputerów osobistych. Druga kategoria to właśnie to, czego brakuje na rynku. Tu, w konkursie może wziąć udział każdy, bez ograniczeń wiekowych. Konkursowe projekty mają być, w założeniu, tanie, szeroko dostępne i łatwe do produkcji. Dlatego też organizatorzy konkursu ustalili maksymalne limity cen składników, służących do produkcji zaprojektowanych urządzeń — do 2 tys. koron w kategorii pierwszej i do 1 tys. w drugiej kategorii. Nagrody konkursowe — pieniężne — są dość wysokie, zwłaszcza dla „młodego portfela", choć nie oszałamiające. Pierwsza nagroda, w obu kategoriach wynosi 4 tys. koron. Dla porównania — przeciętna płaca w Czechosłowacji zbliża się do poziomu 3 tysięcy. Przy ocenie prac bierze się pod uwagę przede wszystkim możliwość praktycznego wykorzystania projektów oraz ich pomysłowość i nowatorstwo. Czy prace konkursowe mają szanse zastosowania w praktyce? Odpowiadając na.to pytanie inż. Vachalik podaje przykłady poprzedniego, ubiegłorocznego konkursu. Większość nagrodzonych projektów już trafiła lub trafi niebawem do produkcji. A były to projekty ciekawe i użyteczne — prosta drukarka, sprzęt Hi-Fi, manipulator kierowany przy pomocy komputera, woltomierz numeryczny.

W ubiegłorocznym konkursie oceniono 71 prac 60 autorów. W tym roku organizatorzy liczą na większe zainteresowanie. Mają chyba ku temu podstawy — przed rokiem nie było jeszcze T-Klubu.

Bolesław K. Jaszczuk



ZŁOTY AMSTRAD

Na wstępie kilka słów o Złotym Stoku. Leży on u wylotu Kotliny Kłodzkiej, w Górach Złotych na granicy z Czechosłowacją.

W naszej miejscowości znajdują się Zakłady Tworzyw i Farb, pracodawca większości mieszkańców. Przy zakładach działa Zakładowy Dom Kultury, w którym jako jedna z wielu sekcji istnieje Klub Mikrokomputerowy. Sprawy, które innym miłośnikom informatyki sprawiły wiele kłopotów tzn. wybór i zakup sprzętu u nas zostały rozwiązane bardzo sprawnie dzięki fachowości pracowników działu informatyki Marka Królikowskiego i Zdzisława Kryczki oraz hojności zakładu, który sfinansował zakup.

Zaczynaliśmy tak jak wszyscy od oswajania się ze sprzętem poprzez gry. Od czerwca tego roku prowadził zajęcia w klubie Włodzimierz Kowalski opiekując się miłośnikami gier komputerowych.

Teraz kilka słów o naszym sprzęcie. Dysponujemy 5 AMSTRADAMI CPC 464, jedną stacją dysków 3 cal., jedną drukarką D-100. Posiadamy również oprogramowanie w postaci kilkunastu programów użytkowych i dydaktycznych, kilkudziesięciu gier (na taśmach i dyskietkach) oraz kilkunastu pozycji książkowych. Ponieważ jesteśmy klubem zakładowym daje to nam możliwość korzystania z Amstradów CPC 6128 oraz sprzętu kompatybilnego z komputerami IBM (również z Winchesterem).

Po kilku miesiącach „łamania palców" zostały rozbudzone kolejne apetyty polegające na zainteresowaniu samodzielnym programowaniem, rozpoczęliśmy więc naukę dwóch najbardziej popularnych języków LOGO i BASIC-u. W zajęciach, które prowadzi Krzysztof Woliński uczestniczą uczniowie szkół podstawowych i średnich oraz dorośli miłośnicy informatyki.

Jak wszyscy również my posiadamy swoje plany. Pragniemy rozpocząć naukę Pascala oraz dBase'a, chcemy również wydawać raz na kwartał naszą gazetkę klubową, w której będziemy podawać najnowsze wieści informatyczne ze świata, z kraju oraz innych klubów.

Interesuje nas nawiązanie współpracy z innymi klubami i prywatnymi użytkownikami Amstradów. Odpiszemy na każdy list i prześlemy pełną listę programów i literatury, którą dysponujemy.

Korespondencję prosimy kierować na adres: **Klub Mikrokomputerowy „Złoty Amstrad" ZDK przy ZTiF, ul. 1-go Maja 10, 57-250 ZŁOTY STOK, woj. wałbrzyskie**.
Kontakt z opiekunem K**lubu Krzysztofem Wolińskim**
praca: ZTiF Dział Informatyki tel. 231 do 236 wew. 242, 253 (w godz. 7-15) dom: Hotel ZDK przy ZTiF tel. 215 (w godz 16-22).

Drogi Bajtku!



Na listy czytelników odpowiada Marcin Waligórski

Jestem opiekunem Szkolnego Koła Mikrokomputerowego o wdzięcznej nazwie „Entruś”. Jest to koło bardzo młode, co pociąga za sobą wiele trudności. Pierwsza to wyposażenie; posiadamy jeden komputer ZX Spectrum 16 KB, który pracuje bez przerwy. Pracę prowadzi się w kilku grupach. Oprogramowanie ze względu na fundusze jest skromne i zamyka się w liczbie kilku gier i kilku programów użytkowych. Pomocą byłby „Bajtek”, który często przedstawia ciekawe programy. Najczęściej są to jednak programy na komputer w wersji 48 KB, a nie 16 KB. Rozumiem, że są olbrzymie trudności w dogodzeniu wszystkim, jednak wierzę, że nie tylko my posiadamy komputer 16 KB.

Drugi problem to budowa wewnętrzna tego komputera, nie mamy np. informacji na temat poszczególnych miejsc pamięci, czy ich adresy są takie same jak w Spectrum 48 KB i czy zwracanie się do nich daje takie same efekty.

Szkoła Podstawowa nr 2
ul. 1000-lecia 89-200
Szubin

Spectrum 48 KB i Spectrum Plus są „młodszymi braćmi” produkowanego od 1982 roku modelu 16 KB. Zmiana wewnętrznej budowy komputera polegała na dodaniu układów pamięci RAM, tworzący dodatkowy blok 32 KB. Nie wymagało to nawet modyfikacji płytki drukowanej komputera, gdyż miejsce dla tych układów zostało uwzględnione już przy projektowaniu Spectrum 16 KB. Z tego powodu rozszerzenie pamięci — równoznaczne przeróbce na Spectrum 48 KB — może być z powodzeniem wykonane w warunkach amatorskich. Swego czasu produkowano nawet tego rodzaju zestawy przeznaczone dla majsterkowiczów.

Dla programisty sytuacja taka oznacza, że zachowane zostały bez zmian: zawartość pamięci ROM (a zatem system operacyjny i BASIC), lokalizacja pamięci obrazu i zmiennych systemowych, sposób komunikowania się z urządzeniami peryferyjnymi itp. W zasadzie jedyną istotną różnicą jest standardowe obniżenie RAMTOP i lokalizacja obszaru grafiki użytkownika poniżej adresu 32767 (a nie jak w modelu 48 KB — 65535, gdyż w omawianym systemie komórki o adresach powyżej 32767 nie istnieją). Ponieważ jednak dostęp do definiowalnych znaków graficznych możliwy jest w obu modelach za pomocą funkcji USR, wynikłe stąd niedogodności jest stosunkowo łatwo usunąć.

Poważne problemy można napotkać w zasadzie w dwóch sytuacjach.

Pierwszą jest brak pamięci na tekst bądź dane programu. W tym przypadku można próbować „zacieśnić” program tak, by zajmował on mniej pamięci, poprzez zmianę jego składni lub stosowanych algorytmów. Możliwości pójścia w tym kierunku są jednak mocno ograniczone. Wprowadzenie dużej ilości tego typu zmian może też odbić się niekorzystnie na szybkości wykonywania programu (aczkolwiek nie musi).

Sytuacja druga to taka, gdy w programie występują odwołania do miejsc pamięci o adresach powyżej 32767. Przykładem mogą być wszelkiego rodzaju nierelokowalne procedury w języku maszynowym. Tutaj możliwości manewru są nieco szersze pod warunkiem znajomości kodu maszynowego procesora Z 80.

W 9/86 numerze „ Bajtka”, w rubryce „Co jest grane” znalazłem opis gry IMPOSSIBLE MISSION.

Zauważyłem, iż brakuje tam kilku mniej lub bardziej ważnych szczegółów.

Pierwszy to ten, że z kawałków (PUZZLE) składa się jeden cały prostokąt. Za tenże prostokąt otrzymuje się jedną z liter 9-literowego hasła. Osobiście grając uzyskałem hasła: NEWCASTLE, VANCOU-VER, SINGAPORE i LIVERPOOL.

Kawałki układanki są zawsze inne — inaczej porozcinany jest prostokąt. Kształt rozcięcia wybiera losowo komputer. Podobnie jest z hasłami, które również wybiera komputer.

Dalej: wszystkich pomieszczeń razem z dwoma, w których znajdują się szachownice, jest 32.

Czas gry wynosi 6 godzin, ale każdy stracony bohater to 10 minut gry mniej. Także sprawdzenie ilości, orientacji i rodzaju PUZZLE powoduje upływ czasu o dwie minuty za każdym sprawdzeniem.

Jeżeli znajdziemy już hasło i doprowadzimy grę do końca, komputer oblicza, ile sekund pozostało do planowego zakończenia gry i w zamian za nie daje nam odpowiednią ilość punktów. Jak grę zakończyć?

Jeżeli z kawałków poskładamy całe prostokąty, zaś uzyskane za nie litery ułożymy w hasło, to meta już blisko. Wiadomo, że w jednej z komnat jest okno. Jeżeli w czasie gry, nie mając skompletowanego hasła, zaczniemy je sprawdzać, otrzymamy komunikat „NOTHING HERE”. Jeżeli jednak hasło jest w naszym posiadaniu i włączamy „se-arching”, następuje zakończenie gry. Jakie? Radzę się przekonać.

Radzę też składać hasło na samym końcu, po znalezieniu już wszystkich elementów i przeszukaniu wszystkich pomieszczeń przedmiotów.

Paweł Andrysiak
ul. Bolesława Chrobrego 128/16
55-200 Oława

Miło nam, że również opisy gier zamieszczane w naszym czasopiśmie, z konieczności dość skrótowe, są obiektem wnikliwej lektury i krytyki. List ten publikuję z dedykacją dla czytelników, którzy mają czas, cierpliwość i ochotę, by „Impossible Mission” ukończyć. Powodzenia.

W jaki sposób z poziomu języka Hisoft Pascal na komputerze Spectrum odczytywać stany urządzeń zewnętrznych: klawiszy klawiatury i joysticka?

Sławomir Stasiak
(adres do wiadomości redakcji)

Odpowiednikiem występującej w BASIC-u funkcji INKEYS jest w Hi-soft-Pascalu standardowa funkcja INCH. Jej wynik jest typu CHAR i wskazuje na aktualnie wciśnięty klawisz. Trudno mi udzielić bliższej informacji na temat komunikacji z drążkiem sterowym, nie



Cześć!

Jestem uczniem klasy 4 szkoły nr 29. Chciałbym uzyskać parę informacji o komputerach: Jakie komputery do 135 tysięcy można kupić w Polsce? Czy opłaca się kupować Spectrum?

Strasznie bym chciał mieć komputer, tata też chce ale nie wie co wybrać.

Wysyłam dla zapoznania swoje zdjęcie. Trochę brzydko wyszedłem.

Przesyłam pozdrowienia dla całej redakcji.

Maciek

Obecnie, poniżej wspomnianej sumy kształtuje się cena giełdowa komputerów Sinclair Spectrum oraz Atari 800XL — ten ostatni jednak wymaga magnetofonu, zatem łączny koszt zestawu może być wyższy.

Centralna Składnica Harcerska oferuje komputery Timex 2048, nowe i z gwarancją, co zależnie od wymagań nabywcy może mieć spore znaczenie.

Czy warto kupować Spectrum..., ta sprawa była już poruszana na ła-

znając typu in-terface'u, którym Pan się posługuje. Zamiast tego podaję kilka mogących się przydać w tym celu funkcji i procedur. Obiekty te są w dialekcie Hisoft Pascal standardowe, tzn. nie należy ich definiować w tekście programu.

PEEK (Adres, Typ) — odczytanie zawartości pamięci począwszy od komórki o adresie Adres. Drugi parametr określa typ wyniku funkcji.

INP (Port) — aktualna zawartość wskazanego portu procesora.

INLINE (Bajt1, Bajt2, BajtN) — umieszczenie w programie pascalowym fragmentu kodu maszynowego.

USER (Adres) — wywołanie podprogramu maszynowego rozpoczynającego się od wskazanego adresu. Podprogram musi kończyć się instrukcją RET procesora (kod 201).

Czy do ZX Spectrum można podłączyć pióro świetlne, myszkę? Jaką drukarkę?

Czy mógłbym gdzieś zakupić ten komputer w cenie nie przekraczającej 150 tys. zł?

(nazwisko i adres do wiadomości redakcji)

Spectrum można wyposażyć zarówno w pióro świetlne, jak i w mysz (firmy AMX). Oba urządzenia wymagają specjalnych interface'ów. Oprogramowanie wykorzystujące te urządzenia nie jest bogate; mysz znalazła jednak efektowne zastosowanie w programie Art Studio.

Najpopularniejszymi i najtańszymi drukarkami przeznaczonymi do współpracy ze Spectrum są ZX Printer (wymagająca metalizowanego papieru) i Seikosha GP-50S. Jakość tych urządzeń jak na aktualne wymagania rynku światowego, nie jest najlepsza. Nie wymagają one natomiast żadnych urządzeń dodatkowych.

Jest możliwe użycie również drukarek innych typów, np. standardu Centronics. Wymaga to jednak z reguły dodatkowego interface'u i transmisji danych przy pomocy dodatkowego programu, tzw. protokołu transmisji.

Centralna Składnica Harcerska ustaliła cenę rozprowadzanych przez siebie komputerów Timex 2048 na 106 tys. złotych; można spodziewać się, że cena ZX Spectrum 48 KB na wolnym rynku nie przekroczy tej granicy. Wersja Spectrum Plus może być kilkanaście procent droższa.

Mam pewien program pisany w BASIC-u (na Spectrum), który nie chce się do końca wgrać z magnetofonu; pod koniec wgrywania pojawia się ERROR. Po „ wylistowaniu" w pewnym momencie pojawiają się znaki zapytania i na końcu dwa migające kursory. Ponieważ program jest długi, nie chciałbym go od nowa pisać, dlatego mam pytanie — jak usunąć podwójny kursor i znaki zapytania, aby pisać program dalej?

Marek Andrys
ul. Stadnina 26
64-410 Sieraków

Jeżeli w wyniku wykonania LIST komputer nie „zawiesił się", to po prostu błędne linie usuwamy — i to jest koniec naszych kłopotów.

W przeciwnym przypadku sytuacja jest znacznie gorsza. Jeżeli wspomniane znaki zapytania (Spectrum wyprowadza w ten sposób niewypisywalne symbole kodu) występują tylko w jednej linii, możemy spróbować usunąć ją bez uprzedniego wypisywania tekstu programu. Domyśli się Pan łatwo, jak to zrobić należy. W momencie wprowadzania lub usuwania linii komputer drukuje automatycznie fragment, w którym dana linia występuje. W razie błędnego wczytania więcej niż jednej linii ryzyko blokady komputera pozostaje.

W tym ostatnim przypadku można próbować ładowania i późniejszej obróbki programu przez inny program, lecz jest to zadanie dość skomplikowane. Prostym rozwiązaniem jest własnoręczne sprowokowanie błędu przez wyłączenie magnetofonu, ZANIM ów fatalny błąd taśmy wystąpi. Oczywiście komputer zasygnalizuje „Tape loading error", lecz taki błąd powinno być już łatwo usunąć sposobami opisanymi powyżej.

mach tej rubryki — i stale powraca w listach Trudno mi odpowiadać na tego rodzaju pytania — zbyt wiele czynników składa się na fakt wyboru komputera, aby można było jednoznacznie stwierdzić: to jest najlepsze, to kupujcie. Spectrum ma w Polsce wiele zalet — oprogramowanie, popularność, największą — jak na razie — ilość wydawanej literatury. Są też wady, o których pisano wielokrotnie (np. częste awarie drogich układów). Uchylając się w ten sposób od jednoznacznej odpowiedzi pragnę jednak przestrzec niedoświadczonych nabywców przed nabywaniem komputerów mało popularnych, bardzo starych lub bardzo prymitywnych — chyba że ktoś rzeczywiście upodoba sobie tego typu urządzenia. Myślę tu o np. ZX-81, Commodore 16 i 116, Atan 600XL, Tandy TRS-80, Meritum I (niestety), Coiour Genie i wielu innych, o których niekiedy nabywca dowiaduje się po raz pierwszy w momencie kupna.

Marcin

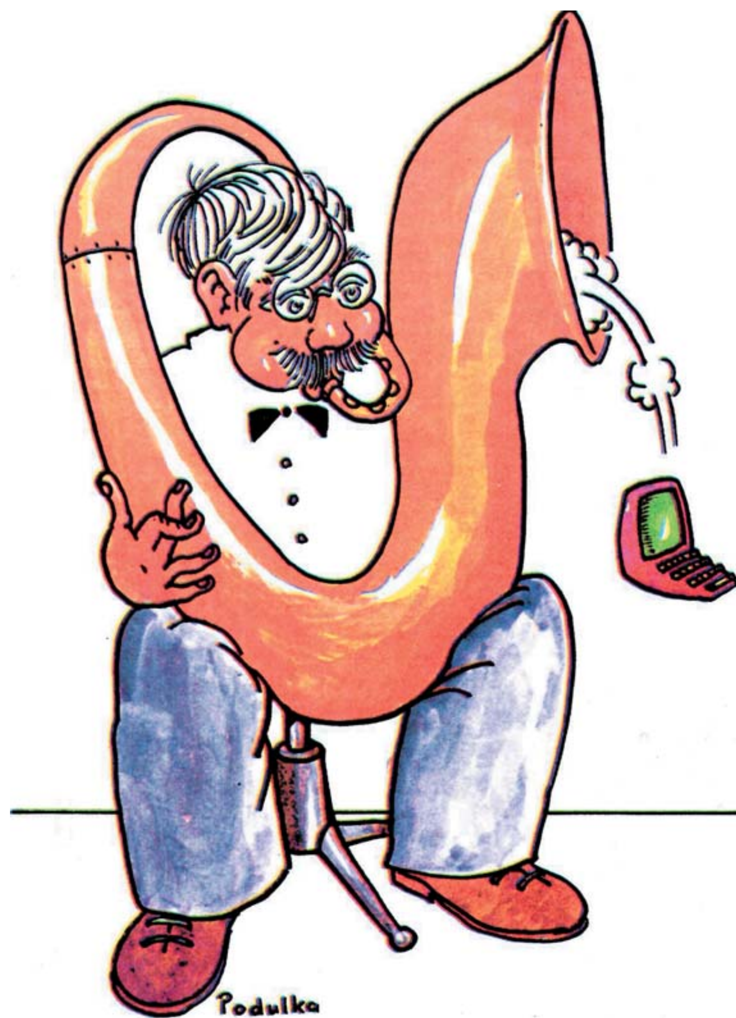
KRUCZKI — I SZTUCZKI

Natura ludzka jest przewrotna. Doskonałym tego przykładem jest właśnie programowanie i komputery. Nic bardziej nie cieszy programisty jak badanie i testowanie możliwości komputera. Z takich testów jednak wynikają czasami rzeczy bardzo zabawne w postaci różnych sposobów i sposobików, które nieraz zdumiewają samych odkrywców.

Chcielibyśmy wobec tego zaproponować wszystkim Czytelnikom „Bajtka" zabawę polegającą na opisaniu i przysłaniu do naszej redakcji takich właśnie kruczków, trików i innych sztuczek. Najlepsze z nich zostaną oczywiście wydrukowane w „Bajtku". Na zakończenie roku wszyscy autorzy opublikowanych trików będą losować nagrodę specjalną.

Nie stawiamy żadnych ograniczeń formalnych i sprzętowych prócz jednego — nie mogą być to sposoby wcześniej opublikowane ani też „wypożyczone" np. z czasopism zachodnich. Listy prosimy kierować na adres:

„Bajtek"
ul.Wspólna 61
00-687 Warszawa
„KRUCZKI I SZTUCZKI"





Wszystkich, którzy nadesłali rozwiązania i zgłoszenia do naszych redakcyjnych konkursów: Świąteczny Konkurs Bajtka, Krzyżówka z dodatkowego numeru „Bajtka", Konkurs MIKRUSEK informujemy, że trwa jeszcze w redakcji sprawdzanie rozwiązań (na zdjęciu). Wyniki konkursów opublikujemy w następnym, numerze „Bajtka".



CIEPŁO I ZIMNO

czyli prezenty od zajączka

Cześć Maluchy!



W każde Święta Wielkanocne odwiedza mnie zajączek (w postaci mamy) i upycha po różnych kątach upominki, których później szukam uważając by żadnego nie przegapić. Gdy byłem młodszy i nie bardzo potrafiłem sobie poradzić z jakimś specjalnie sprytnie schowanym podarkiem mama-zajączek podpowiadała: „Ciepło..., jeszcze cieplej..., gorąco!”

I to jest właśnie pomysł na dzisiejszą zabawę z komputerem — gra dla dwóch osób. Mamy trzydzieści ustawionych obok siebie schowków. Pierwsza z grających osób (zajączek) ukrywa w nich kolejno 5 przedmiotów. Drugi gracz ma znaleźć jak najwięcej prezentów. Ma jednak na to tylko dziesięć prób. Następnie role się zmieniają i teraz szukający bawi się w zajączka.

Do roboty więc. Zobaczmy jak działa nasz program. Na początek — tradycyjnie — określimy sobie tablice i zmienne (linie: 100-140) z których będziemy korzystać w programie. Tablica **schowek** posłuży nam do zapamiętania gdzie zajączek ukrył prezenty — jeśli np. **schowek** (5) = 1 to znaczy, że w piątej skrytce jest upominek. Podobnie jest z tablicą **przykrywka**, z tą różnicą, że jedynka oznacza tu otwartą kryjówkę. Zmienne **pozycja i numer** oznaczają właściwie to samo — numer schowka. Pierwsza używana jest podczas szukania, druga podczas chowania prezentów. Zmienna **znaleziono** praktycznie nie wymaga komentarza, jej wartość określa liczbę znalezionych prezentów.

Linie 200-250 to fragment programu dla zajączka. Komputer czeka na podanie pięciu różnych (linie 220 i 230) liczb, które zostaną zapamiętane jako numery skrytek. Gdy otrzyma oczekiwane informacje przychodzi kolej na rozmowę z poszukiwaczem prezentów (pętla od 0 do 10 w liniach 300-540).

Podczas pierwszego obiegu pętli (p = 0 w linii 310) komputer pomija odczytywanie klawiatury (linie 320 i 330) i przechodzi od razu do rysowania schowków. Liczby na górze ułatwiają określenie numeru kryjówki. Po zrobieniu rysunku sprawdza natomiast czy w ostatnio otwartej skrytce — za pierwszym razem jest to zawsze pierwsza skrytka — nie ma przypadkiem upominku i czy nie znajduje się on w którejś z sąsiednich. Jeśli prezent jest tuż obok pojawia się komunikat „gorąco”, jeśli natomiast o jedną dalej — „ciepło” (linie: 490-510). Niezależnie od tego otrzymujemy informację ile już razy próbowaliśmy zgadywać i ile prezentów znaleźliśmy (linie 520 i 530).

W następnych obiegach pętli komputer za każdym razem będzie oczekiwał decyzji, którą skrytkę chcemy otworzyć (linie 320 i 230). Zwróćcie uwagę w jaki sposób program zabezpieczony jest przed wprowadzeniem liczby, która nie może być numerem schowka, lub przed ponownym otwarciem tego samego schowka.

Jeśli w trakcie dziesięciu prób nie uda nam się znaleźć wszystkich upominków, komputer przechodzi do linii 1000 i melduje nam ile straciliśmy. Jeśli jednak uda nam się zdobyć wszystkie prezenty wówczas wyskakuje z pętli do linii 3000.

Praktycznie cały nasz program skonstruowany jest przy pomocy instrukcji warunkowych IF ... THEN (Użytkownicy mikrokomputerów Amstrad/ /Schneider i Meritum mogą znacznie uprościć niektóre fragmenty programu stosując instrukcję ELSE) oraz operacji logicznych OR i AND. Warto więc dokładnie prześledzić działanie lini: 220, 230, 330, 490-510. Jeśli już to zrobicie spróbujcie napisać program rozwiązujący następujący problem:

> Kubuś z Barnabą wybrali się na wystawę komputerów mając pewną sumę pieniędzy na bilety. Cena normalnego biletu wynosiła 50 zł, ulgowego (dla dzieci poniżej 13 lat) — 25 zł. Przedszkolaki — do sześciu lat włącznie — mogły wchodzić bez biletu, ale wszystkie dzieci, które nie skończyły 10 lat musiały mieć opiekuna w wieku co najmniej 15 lat. Wasz program ma zapytać chłopców ile mają pieniędzy i w jakim są wieku, a następnie odpowiedzieć, czy dostaną się na wystawę.

Najciekawsze (i najkrótsze) rozwiązania wydrukujemy w „Bajtku”. Przysyłając program nie zapomnijcie o podaniu swojego wieku a na kopercie zróbcie dopisek „Sam programuję”.

Kubuś i Barnaba czekają niecierpliwie.

Romek





Instrukcje CLS w liniach 100 i 340 oznaczają kasowanie ekranu w mikrokomputerach Spectrum, Amstrad/ Schneider, Meritum. W innych typach mikrokomputerów należy zastosować instynkcje spełniające te same funkcje.

```
10 REM ************************************
20 REM ********* cieplo zimno *************
30 REM ************************************
99 REM ********* inicjalizacja ************
100 CLS
110 DIM schowek(32)
120 DIM przykrywka(32)
130 LET pozycja=1
140 LET znaleziono=0
199 REM ****** ukrywanie prezentow *******
200 FOR i=1 TO 5
210  INPUT numer
220  IF numer<2 OR nuuer>30 THEN GOTO 210
225  IF schowek(numer)=1 THEN GOTO 210
230  IF schowek(numer)=1 THEN GOTO 210
240   LET schowek (numer) =1
250 NEXT i
299 REM ****** szukanie prezentow *******
300 FOR p=0 TO 10
310  IF p=0 THEN GOTO 340
319  REM **** odczytywanie klawiatury ****
320  INPUT pozycja
330  IF pozycja<>INT (pozycja) OR pozycja<1 OR
     pozycja>30 OR przykrywka(pozycja)=1 THEN
     GOTO 320
339  REM ************ rysunek ************
340  CLS
350  PRINT "         (   1    )(   2    )3"
360  PRINT "123456739012345678901234567890"
370  LET przykrywka(pozycja)=1
380   FOR i=1 TO 30
390    IF przykrywka(i)=0 THEN PRINT "#";
400    IF przykrywka(i)=1 THEN
       IF schowek(i)=1 THEN PRINT "*"
410  IF przykrywka(i)=1 THEN
     IF schowek(i)=0 THEN PRINT " ";
420   NEXT i
430  PRINT
440   FOR i=1 TO 30
450    IF i=pozycja THEN PRINT "^";
460    IF i=pozycja THEN PRINT " ";
470   NEXT i
480  PRINT
489  REM ******* cieplo czy goraco *******
490  IF schowek(pozycja)=1 THEN GOSUB 2000
500  IF (schowek(pozycja-1)=1 AND przykrywka
     (pozycja-1)=0) OR (schowek(pozycja+1)=1
     AND przykrywka(pozycja+1)=0) THEN PRINT
     "Goraco"
510  IF (schowek(ABS(pozycja-2))=1 AND
     przykrywka(ABS(pozycja-2))=0) OR
     (schowek(pozycja+2)=1 AND przykrywka
     (pozycja+2)=0) THEN PRINT "Cieplo"
519  REM ******** komentarze ************
520  PRINT "Proba ";p
530  PRINT "Znalazles";znaleziono; "prezentow."
540 NEXT p

999  REM ****** koniec po 10 probach *******
1000 PRINT "Nie znalazles ";5-znaleziono; "
     prezentow."
1010 END

1999  REM ****** znaleziono *******
2000  PRINT "Jest!!!"
2010  LET znaleziono=znaleziono+1
2020  IF znaleziono=5 THEN GOTO 3000
2030  RETURN

2999  REM ****** sukces *******
3000  PRINT "Znalazles wszystko."
3010  END
```

MIKRUSEK

Kilkaset listów, ponad tysiąc różnych programów przysłali przedszkolacy na konkurs „MIKRUSEK". Wszystkie trzeba dokładnie przeczytać i sprawdzić. Musicie więc jeszcze nieco poczekać na ogłoszenie wyników. Tymczasem prezentujemy propozycje Grzegorza Rowińskiego (lat 11) z Warszawy.

Grzegorz przysłał dwa programy napisane na mikrokomputerze Amstrad. Pierwszy z nich (po prawej) jest programem muzycznym i daje nam możliwość wysłuchania czterech kolęd. Nie byłoby w tym pewnie nic ciekawego — program wykorzystuje zaledwie ułamek możliwości muzycznych Am-strada — gdyby nie fakt, że Grzegorz bardzo sprytnie wykorzystał linie DATA z podobnego programu na Spectrum.

Programowanie dźwięku w Spectrum i Amstra-dzie odbywa się na całkowicie różnych zasadach. Okazuje się jednak, że komputer w rękach sprytnego programisty może sobie poradzić także z takim kłopotem. Tajemnica tkwi w liniach 70 i 80, w których następuje tłumaczenie muzycznego zapisu Spectrum na zrozumiały dla Amstrada.

Drugi program Grzegorza (poniżej) służy do sortowania — układania w kolejności alfabetycznej dowolnych słów. Co prawda przyjęty algorytm sortowania wymaga wykonania dużej liczby operacji i skutkiem tego, sortowanie większych zbiorów trwa dość długo. Niemniej jednak program działa poprawnie i może być przydatny chociażby ze względu na swą prostotę. Przede wszystkim zaś jest to program napisany samodzielnie.

Niestety, nie wszyscy mają ochotę korzystać z własnej głowy. (Uważają chyba że służy ona tylko do noszenia czapki.) Przepisują programy z różnych czasopism i chwalą się, że sami je wymyślili. Absolutnym rekordzistą w tej dziedzinie (przynajmniej na razie) okazał się Artur z Jeżowa Sudeckiego, który przepisał program z "Bajtka". Drogi Arturze, ja naprawdę czytam „Bajtka", i to kilka razy na różnych etapach jego produkcji. A w ogóle to fe, nieładnie...

Romek

```
9 REM ***** WPROWADZANIE DANYCH *****
10 PRINT "Jle slow podasz ";:INPUT l
20 l=l-1
30 DIM a$(l)
40 FOR s=0 TO l
50 PRINT "Podaj ";s+1;" slowo";:INPUT a$(s)
60 NEXT s
99 REM ***** PROGRAM GLOWNY *****
100 FOR t=1 TO l
110 w=0
120 IF a$(t)<=a$(w)THEN GOSUB 150
130 w=w+1
140 IF w=t THEN NEXT t:GOTO 300 ELSE GOTO 120
150 p$=a$(t)
160 FOR z=(t-1) TO w STEP -1
170 a$(z+1)=a$(z)
180 NEXT z
190 a$(w)=p$
200 w=t-1
210 RETURN
299 REM *** WYDRUK DANYCH KONCOWYCH ***
300 FOR a=0 TO l
310 PRINT a$(a)
320 NEXT a
```

```
10 GOSUB 120
20 PRINT "PODAJ ILOSC ZWROTEK  ";:INPUT p
30 FOR z=1 TO p
40 READ t,e
50 FOR n=1 TO e
60 READ a,b
70 SOUND 1,ROUND(62500/(440*(2^(2+((b-10)/12)
)))),t/a*20
80 SOUND 2,ROUND(62500/(440*(2^((-1)+((b-
10)/12))))),t/a*20
90 SOUND 7,500,1,0
100 NEXT n,z
110 GOTO 10
120 INK 0,0:INK 1,15: INK 2,17:PEN 1:MODE 1
130 PRINT "                        ,"
140 PRINT "    ";CHR$(164);" 1986 GRZEGORZ
ROWINSKI"
150 PEN 2:PRINT
160 PRINT "      MENU:":PRINT
170 PRINT "(1) PRZYBIEZELI DO BETLEJEM PASTERZE"
180 PRINT "(2) GDY SIE CHRYSTUS RODZI"
190 PRINT "(3) BOG SIE RODZI"
200 PRINT "(4) LULAJZE,JEZUNIU,MOJA PERELKO"
210 PEN 1:PRINT
220 PRINT "CO WYBIERASZ";:INPUT w
230 ON w GOSUB 250,260,270,280
240 RETURN
250 RESTORE 300:RETURN
260 RESTORE 400:RETURN
270 RESTORE 500:RETURN
280 RESTORE 600:RETURN
300 DATA 10,62
310 DATA 8,3,8,2,8,3,8,5,8,7,8,5,8,7,8,8,4,10,
4,12,2,10
320 DATA 8,3,8,2,8,3,8,5,8,7,8,5,8,7,8,8,4,10,
4,12,2,10
330 DATA 4,15,8,10,8,10,8,12,8,10,8,8,8,7,4,8,8,
8,8,8,8,10,8,8,8,7,8,5,4,7,4,8,2,10,4,7,4, 5,2,3
340 DATA 4,15,8,10,8,10,8,12,8,10,8,8,8,7,4,8,8,
8,8,8,8,10,8,8,8,7,8,5,4,7,4,8,2,10,4,7,4,5,2,3
400 DATA 10,56
410 DATA 4,9,4,6,4,14,4,11,2,11,2,9,4,6,4,2,4,7
,4,4,2,4,2,2
420 DATA 4,9,4,6,4,14,4,11,2,11,2,9,4,6,4,2,4,7
,4,4,2,4,2,2
430 DATA 4,4,4,4,4,7,4,4,4,6,4,6,4,9,4,6,4,4,4,4
,4,7,4,4,4,6,4,6,4,9,4,6,3,9,8,9,2,9,3,11,8,11,2
,11,3,13,8,13,2,13,4,14,4,9,4,11,4,7
,2,6,2,4,1,2
500 DATA 20,70
510 DATA 8,3,8,15,4,15,8,14,8,12,8,12,8,12,4,1
2,8,10,8,8,8,7,8,10,8,15,8,10,8,8,8,7,4,7,4,5,
4,1000
520 DATA 8,3,8,15,4,15,8,14,8,12,8,12,8,12,4,1
2,8,10,8,8,8,7,8,10,8,15,8,10,8,8,8,7,4,7,4,5,
4,1000
530 DATA 3,
8,8,5,8,7,8,8,8,10,8,10,4,7,4,10,3,8,8,5,8,2,8,-
1, 8,3,8,3,4,7,4,10,3,8,8,5,8,7,8,8,8,10,8,10,4,
7,4,10,3, 8,8,5,8,7,8,8,8,10,8,2,4,5,4,3
600 DATA 15,74
610 DATA 4,6,4,6,4,6,4,9,8,7,8,6,4,7,4,4,4,6,4,
7,4,11,2,9,4,6,4,6,4,6,4,9,8,7,8,6,4,7,4,4,3,9,
8,7,4,7,2,6
620 DATA 4,9,4,9,8,14,8,13,4,11,4,11,8,11,8,9,4,
7,4,7,4,7,4,11,2,9,4,6,4,6,4,6,4,9,8,7,8,6,4,7,4
,4,3,9,8,7,4,7,2,6
630 DATA 4,9,4,9,8,14,8,13,4,11,4,11,8,11,8,9,4,
7,4,7,4,7,4,11,2,9,4,6,4,6,4,6,4,9,8,7,8,6,4,7,4
,4,3,9,8,7,4,7,2,6
```

NIE TYLKO KOMPUTERY

SŁONECZNA ZUPA

Dokończenie ze str. 32

Inżynier Vinay Prakash — chemik zatrudniony przez BHEL opowiadał jak na początku z niedowierzaniem dotykano przesuwających się za szklaną szybą obrazów. Dopatrywano się czarów. Bo w jakiż inny sposób żywych poruszających się ludzi można było pomniejszyć i uwięzić we wnętrzu małego pudełka... Dziś telewizor odbierający z satelity program w zrozumiałym powszechnie języku hindi nie jest już sensacją. Choć reakcje tych prostych ludzi przeniesionych raptem do innej epoki są zaskakujące 65-letni Khuda Baksh o niezwykle spracowanych zrogowaciałych dłoniach, przy-dybany przez nas z radłem na ramieniu powiedział: „Dobrze jest popatrzeć na program telewizyjny, ale mnie to specjalnie nie zajmuje..." 13-letni Isrer zafascynowany jest natomiast ruchem, przemieszczającymi się kolorami. Najbardziej przyciągają go sceny taneczne...

Dla nas polskich dziennikarzy ten egzotyczny dom kultury to dopiero przedsmak niezwykłości. Zaproszeni do serca Banbharija odkrywamy nagle wśród glinianych chałupinek... jarzeniówki ustawione na całkiem europejskich słupach z prawdziwego aluminium. Nad każdym słupem zainstalowano płaskie baterie słoneczne składające się z kilkunastu okrągłych zielonkawych fotokomórek. Po zmroku sioło od niepamiętnych czasów oświetlane nędznymi płomykami oliwnych kaganków zalewają strumienie białawej poświaty. Wszystko usytuowano tak, aby czyjeś skore do zabawy ręce nie rozmontowały precyzyjnej instalacji. Na początku oczywiście zdarzały się tego rodzaju psoty. Przecież niezwykle frapujące druty i kable aż zachęcały do manipulacji. Ale po kilku doraźnych naprawach technicy z fabryki zapowiedzieli, że cała odpowiedzialność za uszkodzenia spadnie na mieszkańców i... zabawy ustały.

Woda tutaj to symbol życia, nieodzowny czynnik płodności natury. Do jej czerpania zaprzęgnięto również słońce. Tym razem trzeba Było stworzyć sporej powierzchni daszek na palach, do którego wtopiono ogniwa słoneczne, aby dostarczyć energii poruszającej silnik pompy zanurzonej w studni. Teraz nie trzeba już skórzanych wiader i kołowrotu napędzanego siłą mięsni zwierząt pociągowych. Wystarczy odkręcić najbardziej konwencjonalny, oczywiście dla nas, kran i podstawić wiadro.

Flirt Banbharija z ogniwami, które wynaleziono kiedyś po to, aby ożywić organizmy krążących nad Ziemią pojazdów nie kończy się na 18 jarzeniówkach, pompie i telewizorze. Naczelnik zgodził się zademonstrować nam pracę najnowszego prezentu BHEL. Na centralny placyk wyniesiono duże płaskie pudło wielkości sporego telewizora. Pokrywa kryła wklęsłe zwierciadło skupiające na matowym, czarnym wnętrzu promienie słońca. Pokryte czernią naczynia doprowadzają do wrzenia zawartą w nich wodę po upływie około 30 minut. W ten sposób można ugotować w świetle naszej dobroczynnej gwiazdy słoneczną zupę w sercu indyjskiego interioru.

A lepiej jeść, w sposób higieniczny przygotowywać pożywienie to dla wieśniaków życ bez ustawicznie trapiących te tereny schorzeń przewodu pokarmowego. Czy wykorzystują tę szansę? Dla każdej chaty przydzielono jedną kuchnię słoneczną wraz ze specjalnymi naczyniami. W sumie rozdano 90 kompletów. Inżynier Prakash jest prawie pewien, ze piece używa nie więcej jak 50—60 gospodarstw domowych. Reszta leży bezczynnie. To skala torującej sobie z trudem drogę cywilizacji. Trudno przecież podejrzewać, że wszyscy bez wyjątku zrezygnują od razu z dawnych przyzwyczajeń. Bo właśnie takie są współczesne Indie. Specyficznie przyrządzony koktajl supernowoczesnych technologii i skrajnego zacofania.

Wojciech Łuczak

SŁONECZNA ZUPA

Najpierw pół godziny lotu Boeingiem 737 Indian Airlines z Delhi do Lakhnau — jednego z głównych miast stanu Uttar Pradesz, a potem przesiadka do automobilu przypominającego konstrukcyjnie nasze dawne Warszawy. Ale tylko takie właśnie pojazdy, wytwarzane całkowicie w Indiach (Hindusi z dumą to podkreślają) wytrzymują dziesiątki tysięcy kilometrów wąskich, asfaltowych, ale niezwykle wyboistych dróg Półwyspu Dekan.

Aby stanąć oko w oko z oszałamiającym technologicznym paradoksem rozwijającego się błyskawicznie kraju, pokonać musieliśmy we wnętrzu wozu Ambasador Mark IV około 90 kilometrów, poruszając się wśród płaskiego jak stół, trochę brunatnego, trochę szarego terenu Niziny Gangesu. Uttar Pradesz nie jest tak bezgranicznie biedny jak pustynny Radżastan, z którego ludzkie szkielety uciekają do wielkich miast, aby w pocie czoła harować w głębokich wykopach za kilkadziesiąt paisa dziennie (100 paisa to jedna rupia, czyli po oficjalnym kursie około 1/13 dolara). Ale do poziomu stopy życiowej przedmieść Madrasu jest tu tak daleko, jak na Księżyc.

Z tego wszystkiego co słyszeliśmy i widzieliśmy można wywnioskować, iż rząd oraz premier osobiście traktują te okolice jako rodzaj poligonu doświadczalnego cywilizacji. Stąd wywodzi się rodzina Ghandich. Rajiv — syn In-diry, dawniej kapitan — pilot pasażerskiego odrzutowca, dziś szef delhijskiego gabinetu, w parlamencie centralnym reprezentuje okręg Amethi niedaleko Jagdishpuru, dokąd właśnie zmierzamy. Podobno starannie wybrano adekwatne do warunków klimatycznych i glebowych modele rozwoju poszczególnych gmin. Postanowiono, że właśnie tam staną fabryki, a gdzie indziej zadba się o podnoszenie wyłącznie kultury rolnej. Na razie to początek traktu mającego wyprowadzić tych, którzy od pokoleń mozolnie zmagając się z ziemią usiłowali jedynie utrzymać się przy życiu, z ciemnoty i zacofania. Czasami jednak analfabetom o ciemnej jak heban skórze umożliwiono tak gwałtowny skok cywilizacyjny przez stulecia, że aż trudno w to wszystko uwierzyć.

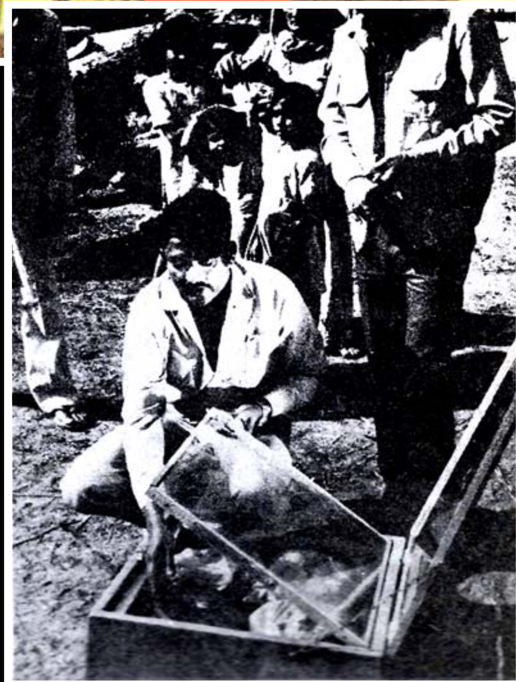
W Jagdishpurze 3 marca 1984 roku Indira Ghandi uroczyście otworzyła podwoje nowoczesnej fabryki izolatorów należącej do ogromnego państwowego kartelu przemysłu elektrycznego Bharat Heavy Electricals Limited znanego już na światowych rynkach jako BHEL. O samym zakładzie można byłoby pisać w nieskończoność. O japońskich liniach technologicznych, o najwyższym standardzie kontroli jakości, o sporych sukcesach eksportowych, o wielce życzliwym Polakom dyrektorze S.K. Gupta i jego sympatycznych, a co ciekawe, niezwykle młodych inżynierach. Najważniejsze wszakże jest to, że zakład, kolonię domków załogi, farmy i tuczarnie dostarczające własnych produktów żywnościowych wzniesiono nie tylko po to, aby Indie przestały importować izolatory do potrzebnych jak woda w tym kraju linii przesyłowych wysokiego napięcia. Fabryka ma być bowiem dla zapóźnionego regionu centrum kultury technicznej. Tworzyć zupełnie nowych w tych stronach standard życia, przyzwyczajać stopniowo wieśniaków do życia nowoczesnego, do obcowania ze zdobyczami cywilizacyjnymi i technologicznymi.

Całkowicie muzułmańską wioskę Banbharija wybrano na chybił trafił. Właściwie niczym nie różniłaby się od zapomnianych przez historię siół, do których dotarliśmy, a gdzie byliśmy pierwszymi oglądanymi przez ich mieszkańców Europejczykami o dziwnej tutaj bladej cerze. Gliniane lepianki kryte liśćmi palmowymi i słomą z gromadami ślicznych, wszędobylskich dzieci taplających się beztrosko w gorącym błocie, byłyby takie same jak wszędzie, gdyby nie eksperyment fabryki w Jagdishapurze.

Tym ludziom mieszkającym pospołu z domowym bydłem postanowiono nagle przybliżyć świat schyłku XX wieku. Najpierw zbudowano wiejskie centrum kultury w postaci murowanego budynku z białej cegły, gdzie na honorowym miejscu ustawiono dar królewski BHEL — kolorowy telewizor. Cóż jednak począć skoro najbliższa linia elektryczna przebiega co najmniej kilkadziesiąt kilometrów od Banbharija. Słońca i wiatru w Indiach jednak nie brakuje. Fabryka sfinansowała instalację baterii słonecznych oraz minielektrowni wiatrowej, które ładują akumulatory, dzięki temu barwne ruchome obrazy stają się własnością wieśniaków. Ich ojcom takie luksusy nie mogły się nawet przyśnić.

Dokończenie na str. 31

OW Banbharija przed wiejskim domem kultury: 65-letni Khuda Baksh i Jego sąsiad na tle jarzeniówki zasilanej bateriami słonecznymi i minielektrowni wietrznej ładującej akumulatory, z których prąd płynie do odbiornika telewizyjnego. Fot. Ireneusz Sobieszczuk



Na centralny placyk wniesiono duże płaskie pudło... inżynier Vinay Prakash demonstruje działanie kuchni słonecznej. Fot. W. Łuczak



www.t2e.pl

Seria „Reduks" ma na celu przetworzenie – pełne zdigitalizowanie prasy komputerowej z początków okresu informatyzacji naszego kraju.

W odróżnieniu od plików w formacie djvu i cbr gdzie format papierowy jest tylko zeskanowany i połączony w jeden dokument, **reduks polega na ponownym przełamaniu dokumentu, wraz z ponowną obróbką graficzną - często z ich ponowną kreacją**. Tam gdzie to możliwe zachowana jest grafika z pierwowzoru, jednak jeśli materiał jest zbyt złej jakości - zostaje on podmieniony (oczywiście jeśli jest dostępny) na równoważny. Jeżeli uda się odnaleźć błąd w zamieszczonych listingach są one poprawiane tak, aby mogły zostać uruchomione na emulatorach lub mikrokomputerach.

Rozpocząłem ten cykl od „Bajtków", jednak mam nadzieję, że uda mi się wzbogacić kolekcję zdigitalizowanych i ponownie przełamanych periodyków także o inne ważne tytuły z lat ubiegłych, o których powoli zapominamy, a bez których współczesny obraz komputeryzacji wyglądałby zgoła inaczej.

Wydania „Reduksowe" są bezpłatne, nie roszczę sobie żadnych praw do zawartości przetwarzanych magazynów, jednak ilość czasu i pracy jaką wkładam aby mogły one trafić w Twoje ręce jest na tyle spora, że pozwolę sobie napomknąć iż możesz wesprzeć mnie przekazując dotację w dowolnej wysokości.

Jeśli posiadasz materiały fotograficzne lub edytorskie, które są lepszej jakości niż prezentowane w numerach, daj znać.



W ramach wolnego czasu odtwarzam stare magazyny komputerowe i prowadzę serwis emulacyjny try2emu. Jeżeli uważasz, że moja publiczna działalność możesz postawić mi kawę, a kawę lubię;)



https://buycoffee.to/t2e

Osoby, które wspierają serwis datkami* mają dostęp do reduksów w wysokiej rozdzielczości z usuniętymi reklamami.

W zależności od wysokości datku każdy otrzymuje czasowy dostęp do reduksów w pełnej rozdzielczości na miesiąc, 3 miesiące lub 6 miesięcy.
Osoby, które aktywnie wspierają projekt mają dostęp nieograniczony.